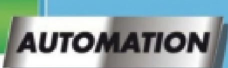
Handbuch





WAGO-I/O-SYSTEM 750

Programmierbarer Feldbuscontroller ETHERNET TCP/IP

750-841

10/100 Mbit/s; digitale und analoge Signale

Version 1.5.1



WAGO Kontakttechnik GmbH & Co. KG

Hansastraße 27
D-32423 Minden

Tel.: +49 (0) 571/8 87 – 0
Fax: +49 (0) 571/8 87 – 1 69

E-Mail: info@wago.com

Web: http://www.wago.com

Technischer Support

Tel.: +49 (0) 571/8 87 – 5 55
Fax: +49 (0) 571/8 87 – 85 55

E-Mail: support@wago.com

Es wurden alle erdenklichen Maßnahmen getroffen, um die Richtigkeit und Vollständigkeit der vorliegenden Dokumentation zu gewährleisten. Da sich Fehler, trotz aller Sorgfalt, nie vollständig vermeiden lassen, sind wir für Hinweise und Anregungen jederzeit dankbar.

E-Mail: documentation@wago.com



Inhaltsverzeichnis











1 Hinweise zu dieser Dokumentation

Hinweis



Dokumentation aufbewahren!
Diese Dokumentation ist Teil des Produkts. Bewahren Sie deshalb die Dokumentation während der gesamten Lebensdauer des Gerätes auf. Geben Sie die Dokumentation an jeden nachfolgenden Besitzer oder Benutzer des Gerätes weiter. Stellen Sie darüber hinaus sicher, dass gegebenenfalls jede erhaltene Ergänzung in die Dokumentation mit aufgenommen wird.

1.1 Gültigkeitsbereich

Die vorliegende Dokumentation gilt für das Gerät:
„Programmierbarer Feldbuscontroller ETHERNET TCP/IP“ 750-841.

Ein Programmierbarer Feldbuscontroller ETHERNET TCP/IP 750-841 darf nur nach Anweisungen dieser Betriebsanleitung und der Systembeschreibung zum WAGO-I/O-SYSTEM 750 installiert und betrieben werden.

ACHTUNG



Versorgungsauslegung des WAGO-I/O-SYSTEM 750 beachten!
Sie benötigen zusätzlich zu dieser Betriebsanleitung die Systembeschreibung zum WAGO-I/O-SYSTEM 750, die unter www.wago.com herunterzuladen ist. Dort erhalten Sie unter anderem wichtige Informationen zu Potentialtrennung, Systemversorgung und Einspeisungsvorschriften.

1.2 Urheberschutz



1.3 Symbole

GEFAHR 	**Warnung vor Personenschäden!** Kennzeichnet eine unmittelbare Gefährdung mit hohem Risiko, die Tod oder schwere Körperverletzung zur Folge haben wird, wenn sie nicht vermieden wird.
GEFAHR 	**Warnung vor Personenschäden durch elektrischen Strom!** Kennzeichnet eine unmittelbare Gefährdung mit hohem Risiko, die Tod oder schwere Körperverletzung zur Folge haben wird, wenn sie nicht vermieden wird.
WARNUNG 	**Warnung vor Personenschäden!** Kennzeichnet eine mögliche Gefährdung mit mittlerem Risiko, die Tod oder (schwere) Körperverletzung zur Folge haben kann, wenn sie nicht vermieden wird.
VORSICHT 	**Warnung vor Personenschäden!** Kennzeichnet eine mögliche Gefährdung mit geringem Risiko, die leichte oder mittlere Körperverletzung zur Folge haben könnte, wenn sie nicht vermieden wird.
ACHTUNG 	**Warnung vor Sachschäden!** Kennzeichnet eine mögliche Gefährdung, die Sachschaden zur Folge haben könnte, wenn sie nicht vermieden wird.
ESD 	**Warnung vor Sachschäden durch elektrostatische Aufladung!** Kennzeichnet eine mögliche Gefährdung, die Sachschaden zur Folge haben könnte, wenn sie nicht vermieden wird.
Hinweis 	**Wichtiger Hinweis!** Kennzeichnet eine mögliche Fehlfunktion, die aber keinen Sachschaden zur Folge hat, wenn sie nicht vermieden wird.
Information 	**Weitere Information** Weist auf weitere Informationen hin, die kein wesentlicher Bestandteil dieser Dokumentation sind (z. B. Internet).



1.4 Darstellung der Zahlensysteme

Tabelle 1: Darstellungen der Zahlensysteme

Zahlensystem	Beispiel	Bemerkung
Dezimal	100	normale Schreibweise
Hexadezimal	0x64	C-Notation
Binär	'100' '0110.0100'	in Hochkomma, Nibble durch Punkt getrennt

1.5 Schriftkonventionen

Tabelle 2: Schriftkonventionen

Schriftart	Bedeutung
kursiv	Namen von Pfaden und Dateien werden kursiv dargestellt z. B.: *C:\Programme\WAGO-I/O-CHECK*
Menü	Menüpunkte werden fett dargestellt z. B.: **Speichern**
>	Ein Größer als Zeichen zwischen zwei Namen bedeutet die Auswahl eines Menüpunktes aus einem Menü z. B.: **Datei > Neu**
Eingabe	Bezeichnungen von Eingabe- oder Auswahlfeldern werden fett dargestellt z. B.: **Messbereichsanfang**
„Wert“	Eingabe- oder Auswahlwerte werden in Anführungszeichen dargestellt z. B.: Geben Sie unter **Messbereichsanfang** den Wert „4 mA“ ein.
[Button]	Schaltflächenbeschriftungen in Dialogen werden fett dargestellt und in eckigen Klammern eingefasst z. B.: **[Eingabe]**
[Taste]	Tastenbeschriftungen auf der Tastatur werden fett dargestellt und in eckigen Klammern eingefasst z. B.: **[F5]**

2 Wichtige Erläuterungen

Dieses Kapitel beinhaltet ausschließlich eine Zusammenfassung der wichtigsten Sicherheitsbestimmungen und Hinweise. Diese werden in den einzelnen Kapiteln wieder aufgenommen. Zum Schutz vor Personenschäden und zur Vorbeugung von Sachschäden an Geräten ist es notwendig, die Sicherheitsrichtlinien sorgfältig zu lesen und einzuhalten.

2.1 Rechtliche Grundlagen

2.1.1 Änderungsvorbehalt

Die WAGO Kontakttechnik GmbH & Co. KG behält sich Änderungen, die dem technischen Fortschritt dienen, vor. Alle Rechte für den Fall der Patenterteilung oder des Gebrauchmusterschutzes sind der WAGO Kontakttechnik GmbH & Co. KG vorbehalten. Fremdprodukte werden stets ohne Vermerk auf Patentrechte genannt. Die Existenz solcher Rechte ist daher nicht auszuschließen.

2.1.2 Personalqualifikation

Sämtliche Arbeitsschritte, die an den Geräten der Serie 750 durchgeführt werden, dürfen nur von Elektrofachkräften mit ausreichenden Kenntnissen im Bereich der Automatisierungstechnik vorgenommen werden. Diese müssen mit den aktuellen Normen und Richtlinien für die Geräte und das Automatisierungsumfeld vertraut sein.

Alle Eingriffe in die Steuerung sind stets von Fachkräften mit ausreichenden Kenntnissen in der SPS-Programmierung durchzuführen.

2.1.3 Bestimmungsgemäße Verwendung der Serie 750

Feldbuskoppler, Feldbuscontroller und Busklemmen des modularen WAGO-I/O-SYSTEM 750 dienen dazu, digitale und analoge Signale von Sensoren aufzunehmen und an Aktoren auszugeben oder an übergeordnete Steuerungen weiterzuleiten. Mit den programmierbaren Feldbuscontrollern ist zudem eine (Vor-)Verarbeitung möglich.

Die Geräte sind für ein Arbeitsumfeld entwickelt, welches der Schutzklasse IP20 genügt. Es besteht Fingerschutz und Schutz gegen feste Fremdkörper bis 12,5 mm, jedoch kein Schutz gegen Wasser. Der Betrieb der Komponenten in nasser und staubiger Umgebung ist nicht gestattet, sofern nicht anders angegeben.

Not-Aus-Funktionen und -Einrichtungen dürfen nicht unwirksam gemacht werden. Beachten Sie dazu die einschlägigen Normen, z. B. die DIN EN 418.

Der Betrieb von Geräten der Serie 750 im Wohnbereich ist ohne weitere Maßnahmen nur zulässig, wenn diese die Emissionsgrenzen (Störaussendungen) gemäß EN 61000-6-3 einhalten. Entsprechende Angaben finden Sie im Kapitel



„Das WAGO-I/O-SYSTEM 750“ → „Systembeschreibung“ → „Technische Daten“ im Handbuch zum eingesetzten Feldbuskoppler/-controller.

Für den Betrieb des WAGO-I/O-SYSTEM 750 in explosionsgefährdeten Bereichen ist ein entsprechender Gehäuseschutz gemäß der Richtlinie 94/9/EG erforderlich. Zusätzlich ist zu beachten, dass eine Baumusterprüfbescheinigung erwirkt werden muss, die den korrekten Einbau des Systems im Gehäuse bzw. Schaltschrank bestätigt.

2.1.4 Technischer Zustand der Geräte

Die Geräte werden ab Werk für den jeweiligen Anwendungsfall mit einer festen Hard- und Software-Konfiguration ausgeliefert. Alle Veränderungen an der Hard- oder Software sowie der nicht bestimmungsgemäße Gebrauch der Komponenten bewirken den Haftungsausschluss der WAGO Kontakttechnik GmbH & Co. KG.

Wünsche an eine abgewandelte bzw. neue Hard- oder Software-Konfiguration richten Sie bitte an die WAGO Kontakttechnik GmbH & Co. KG.

2.2 Sicherheitshinweise

Beim Einbauen des Gerätes in Ihre Anlage und während des Betriebes sind folgende Sicherheitshinweise zu beachten:

GEFAHR



Nicht an Geräten unter Spannung arbeiten!
Schalten Sie immer alle verwendeten Spannungsversorgungen für das Gerät ab, bevor Sie es montieren, Störungen beheben oder Wartungsarbeiten vornehmen.

GEFAHR



Einbau nur in Gehäusen, Schränken oder elektrischen Betriebsräumen!
Das WAGO-I/O-SYSTEM 750 mit seinen Geräten ist ein offenes Betriebsmittel. Bauen Sie dieses ausschließlich in abschließbaren Gehäusen, Schränken oder in elektrischen Betriebsräumen auf. Ermöglichen Sie nur autorisiertem Fachpersonal den Zugang mittels Schlüssel oder Werkzeug.

GEFAHR



Unfallverhütungsvorschriften beachten!
Beachten Sie bei der Montage, Inbetriebnahme, Wartung und Störbehebung die für Ihre Maschine zutreffenden Unfallverhütungsvorschriften wie beispielsweise die BGV A 3, „Elektrische Anlagen und Betriebsmittel".

GEFAHR



Auf normgerechten Anschluss achten!
Zur Vermeidung von Gefahren für das Personal und Störungen an Ihrer Anlage, verlegen Sie die Daten- und Versorgungsleitungen normgerecht und achten Sie auf die korrekte Anschlussbelegung. Beachten Sie die für Ihre Anwendung zutreffenden EMV-Richtlinien.

ACHTUNG



Defekte oder beschädigte Geräte austauschen!
Tauschen Sie defekte oder beschädigte Geräte (z. B. bei deformierten Kontakten) aus, da die Funktion der betroffenen Geräte langfristig nicht sichergestellt ist.

ACHTUNG



Geräte vor kriechenden und isolierenden Stoffen schützen!
Die Geräte sind unbeständig gegen Stoffe, die kriechende und isolierende Eigenschaften besitzen, z. B. Aerosole, Silikone, Triglyceride (Bestandteil einiger Handcremes). Sollten Sie nicht ausschließen können, dass diese Stoffe im Umfeld der Geräte auftreten, bauen Sie die Geräte in ein Gehäuse ein, das resistent gegen oben genannte Stoffe ist. Verwenden Sie generell zur Handhabung der Geräte saubere Werkzeuge und Materialien.



ACHTUNG



Reinigung nur mit zulässigen Materialien!
Reinigen Sie verschmutzte Kontakte mit ölfreier Druckluft oder mit Spiritus und einem Ledertuch.





ACHTUNG



Kein Kontaktspray verwenden!
Verwenden Sie kein Kontaktspray, da in Verbindung mit Verunreinigungen die Funktion der Kontaktstelle beeinträchtigt werden kann.

ACHTUNG



Verpolungen vermeiden!
Vermeiden Sie die Verpolung der Daten- und Versorgungsleitungen, da dies zu Schäden an den Geräten führen kann.

ESD



Elektrostatische Entladung vermeiden!
In den Geräten sind elektronische Komponenten integriert, die Sie durch elektrostatische Entladung bei Berührung zerstören können. Beachten Sie die Sicherheitsmaßnahmen gegen elektrostatische Entladung gemäß DIN EN 61340-5-1/-3. Achten Sie beim Umgang mit den Geräten auf gute Erdung der Umgebung (Personen, Arbeitsplatz und Verpackung).

3 Systembeschreibung

Das WAGO-I/O-SYSTEM 750 ist ein modulares und feldbusunabhängiges E/A-System. Der hier beschriebene Aufbau besteht aus einem Feldbuskoppler/-controller (1) und den angereihten Busklemmen (2) für beliebige Signalformen, die zusammen den Feldbusknoten bilden. Die Endklemme (3) schließt den Knoten ab.

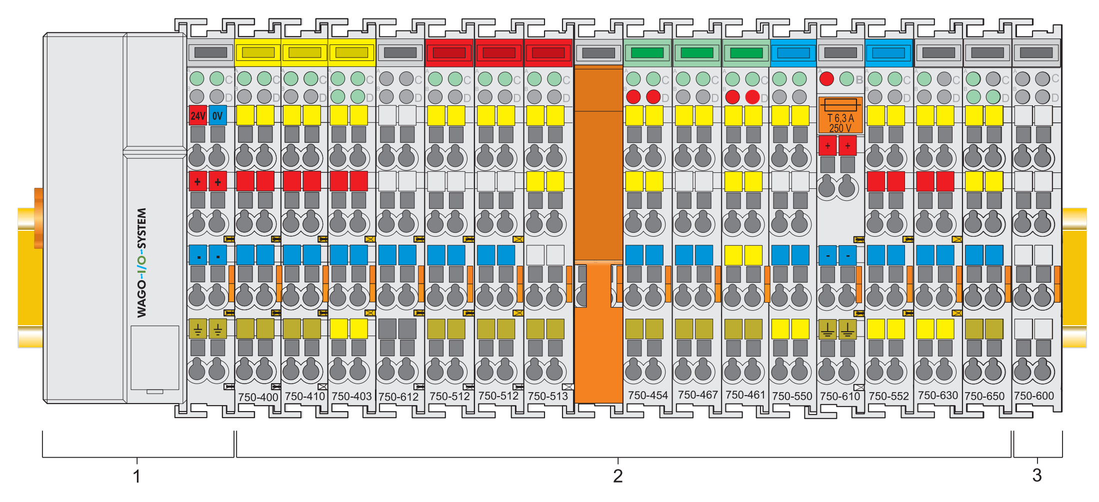


Abbildung 1: Feldbusknoten

Feldbuskoppler/-controller stehen für diverse Feldbussysteme zur Verfügung.

Standard-Feldbuskoppler/-controller und erweiterte ECO-Feldbuskoppler enthalten ein Feldbus-Interface, eine Elektronik und eine Einspeiseklemme. Das Feldbus-Interface bildet die physikalische Schnittstelle zum jeweiligen Feldbussystem. Die Elektronik verarbeitet die Daten der Busklemmen und stellt diese für die Feldbuskommunikation bereit. Über die integrierte Einspeiseklemme werden die 24V-Systemversorgung und die 24V-Feldversorgung eingespeist.

Der Feldbuskoppler/-controller kommuniziert über den jeweiligen Feldbus. Die programmierbaren Feldbuscontroller (PFC) ermöglichen zusätzlich SPS-Funktionen zu implementieren. Die Programmierung erfolgt mit WAGO-I/O-*PRO* CAA gemäß IEC 61131-3.

An den Feldbuskoppler/-controller können Busklemmen für unterschiedliche digitale und analoge E/A-Funktionen sowie Sonderfunktionen angereiht werden. Die Kommunikation zwischen Feldbuskoppler/-controller und Busklemmen erfolgt über einen internen Bus, den Klemmenbus.

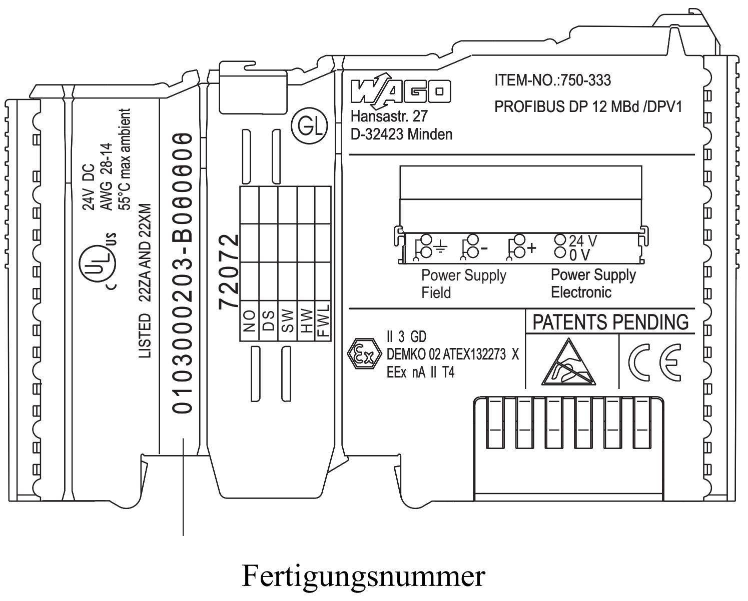
Die Komponenten des WAGO-I/O-SYSTEM 750 besitzen eine übersichtliche Anschlussebene, Leuchtdioden für die Statusanzeige, einsteckbare Mini-WSB-Schilder und herausziehbare Gruppenbezeichnungsträger.

Die 3-Leitertechnik, ergänzt durch einen Schutzleiteranschluss, erlaubt eine direkte Sensor-/Aktorverdrahtung.



3.1 Fertigungsnummer

Die Fertigungsnummer gibt den Auslieferungszustand direkt nach Herstellung an. Diese Nummer ist Teil der seitlichen Bedruckung jeder Komponente. Zusätzlich wird die Fertigungsnummer auf die Abdeckklappe der Konfigurations- und Programmierschnittstelle des Feldbuskopplers/-controllers gedruckt.



01	03	00	02	03	- B000000
Kalender-woche	Jahr	Software-Version	Hardware-Version	Firmware-Loader-Version	Interne Nummer

Abbildung 2: Beispiel einer Fertigungsnummer

Die Fertigungsnummer setzt sich zusammen aus Herstellwoche und -jahr, Software-Version (optional), Hardware-Version, Firmware-Loader-Version (optional) und weiteren internen Informationen der WAGO Kontakttechnik GmbH & Co. KG.

3.2 Hardware-Adresse (MAC-ID)

Ein Programmierbarer Feldbuscontroller ETHERNET TCP/IP hat immer eine einmalige und weltweit eindeutige physikalische Adresse, die MAC-ID (Media-Access-Control-Identity). Diese befindet sich auf der Rückseite des Controllers sowie auf einem selbstklebenden Abreiß-Etikett auf der Seite des Controllers. Die MAC-ID besitzt eine feste Länge von 6 Byte (48 Bit) in hexadezimaler Schreibweise. Die ersten 3 Byte dienen der Herstellerkennung (z. B. 00:30:DE für WAGO). Die weiteren 3 Byte geben die laufende Seriennummer für die Hardware an.

3.3 Komponenten-Update

Für den Fall des Updates einer Komponente, enthält die seitliche Bedruckung jeder Komponente eine vorbereitete Matrix.

Diese Matrix stellt für insgesamt drei Updates Spalten zum Eintrag der aktuellen Update-Daten zur Verfügung, wie Betriebsauftragsnummer (NO; ab KW 13/2004), Update-Datum (DS), Software-Version (SW), Hardware-Version (HW) und die Firmware-Loader-Version (FWL, optional).

Aktuelle Versionsangabe für		1. Update	2. Update	3. Update	
BA-Nummer	**NO**				← ab KW 13/2004
Datestamp	**DS**				
Software-Index	**SW**				
Hardware-Index	**HW**				
Firmwareloader-Index	**FWL**				← nur Koppler/Controller

Ist das Update einer Komponente erfolgt, werden die aktuellen Versionsangaben in die Spalten der Matrix eingetragen.

Zusätzlich wird bei dem Update eines Feldbuskopplers/-controllers auch die Abdeckklappe der Konfigurationsschnittstelle mit der aktuellen Fertigungs- und Betriebsauftragsnummer bedruckt.

Die ursprünglichen Fertigungsangaben auf dem Gehäuse der Komponente bleiben dabei erhalten.

3.4 Lagerung, Kommissionierung und Transport

Die Komponenten sind möglichst in der Originalverpackung zu lagern. Ebenso bietet die Originalverpackung beim Transport den optimalen Schutz.

Bei Kommissionierung oder Umverpackung dürfen die Kontakte nicht verschmutzt oder beschädigt werden. Die Komponenten müssen unter Beachtung der ESD-Hinweise in geeigneten Behältern/Verpackungen gelagert und transportiert werden.



3.5 Aufbaurichtlinien und Normen

DIN 60204	Elektrische Ausrüstung von Maschinen
DIN EN 50178	Ausrüstung von Starkstromanlagen mit elektronischen Betriebsmitteln (Ersatz für VDE 0160)
EN 60439	Niederspannung – Schaltgerätekombinationen



3.6 Spannungsversorgung

3.6.1 Potentialtrennung

Innerhalb des Feldbusknotens bestehen drei galvanisch getrennte Potentialgruppen:

- galvanisch getrenntes Feldbusinterface mittels Übertrager
- Elektronik des Feldbuskopplers/-controllers und der Busklemmen (Klemmenbus)
- Alle Busklemmen besitzen eine galvanische Trennung zwischen der Elektronik (Klemmenbus, Logik) und der feldseitigen Elektronik. Bei einigen digitalen und analogen Eingangsklemmen ist diese Trennung kanalweise aufgebaut, siehe Katalog.

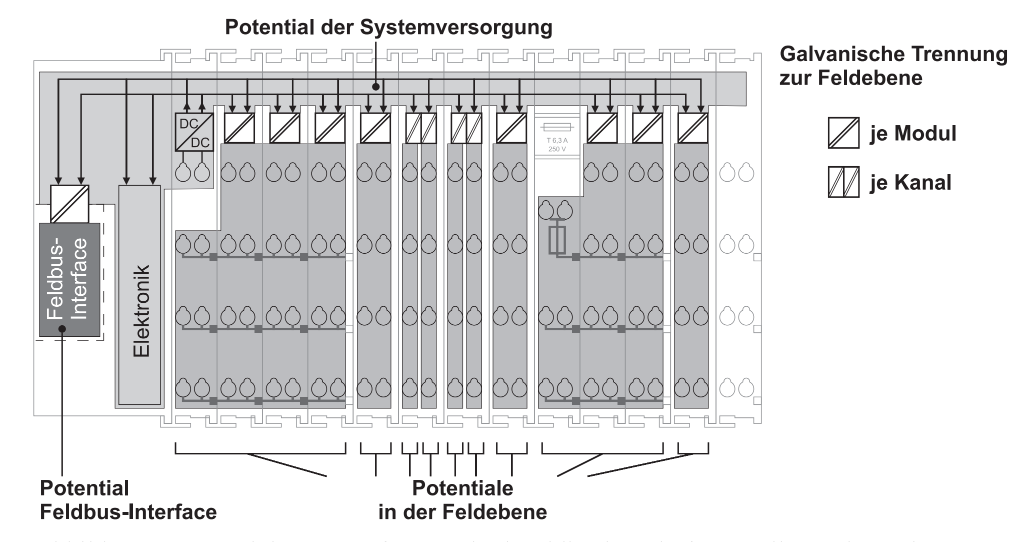


Abbildung 3: Potentialtrennung für Standard-Feldbuskoppler/-controller und erweiterte ECO-Feldbuskoppler

Hinweis



Schutzleiterfunktion sicherstellen (evtl. durch Ringspeisung)!
Beachten Sie, dass der Schutzleiteranschluss in jeder Gruppe vorhanden sein muss. Damit unter allen Umständen die Schutzleiterfunktion erhalten bleibt, kann es sinnvoll sein, wenn Sie den Anschluss am Anfang und Ende einer Potentialgruppe auflegen (siehe Kapitel „Erdung“ > „Schutzerde", Ringspeisung). Sollten Sie bei Wartungsarbeiten eine Busklemme aus dem Verbund lösen, ist dadurch der Schutzleiteranschluss für alle angeschlossenen Feldgeräte gewährleistet.
Verwenden Sie ein gemeinsames Netzteil für die 24V-Systemversorgung und die 24V-Feldversorgung, so wird die galvanische Trennung zwischen Klemmenbus und Feldebene für die Potentialgruppe aufgehoben.

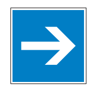


3.6.2 Systemversorgung

3.6.2.1 Anschluss

Das WAGO-I/O-SYSTEM 750 benötigt als Systemversorgung eine 24V-Gleichspannung (-15% / +20 %). Die Einspeisung erfolgt über den Feldbuskoppler/-controller und bei Bedarf zusätzlich über die Potentialeinspeiseklemmen mit Busnetzteil (750-613). Die Einspeisung ist gegen Verpolung geschützt.

Hinweis



Keine unzulässige Spannung/Frequenz aufschalten!
Schalten Sie keine unzulässigen Spannungs- oder Frequenzwerte auf, dieses kann zur Zerstörung der Baugruppe führen.

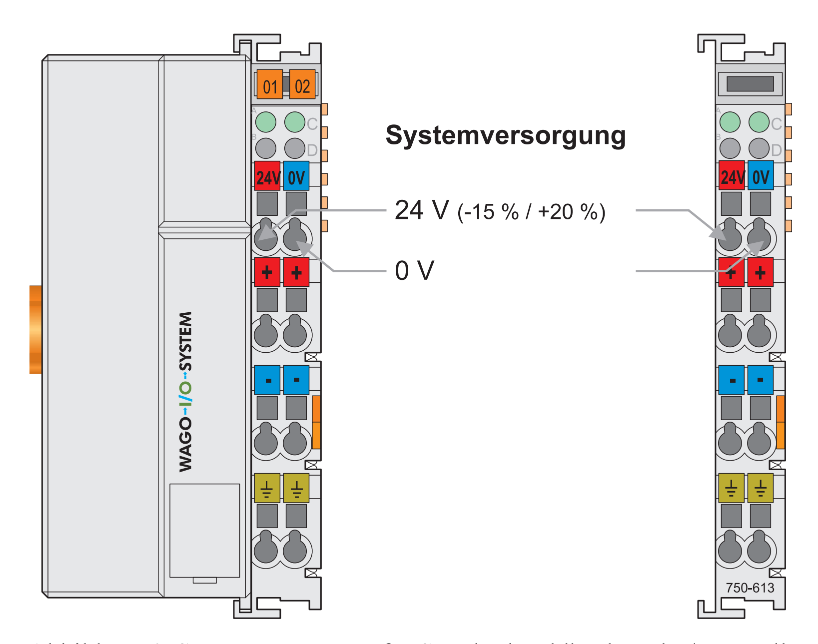


Abbildung 4: Systemversorgung für Standard-Feldbuskoppler/-controller und erweiterter ECO-Feldbuskoppler

Die eingespeiste 24 V-Gleichspannung versorgt alle systeminternen Bausteine, z. B. die Elektronik des Feldbuskopplers/-controllers, das Feldbus-Interface und die Busklemmen über den Klemmenbus (5 V-Systemspannung). Die 5 V-Systemspannung ist mit der 24 V-Systemversorgung galvanisch verbunden.

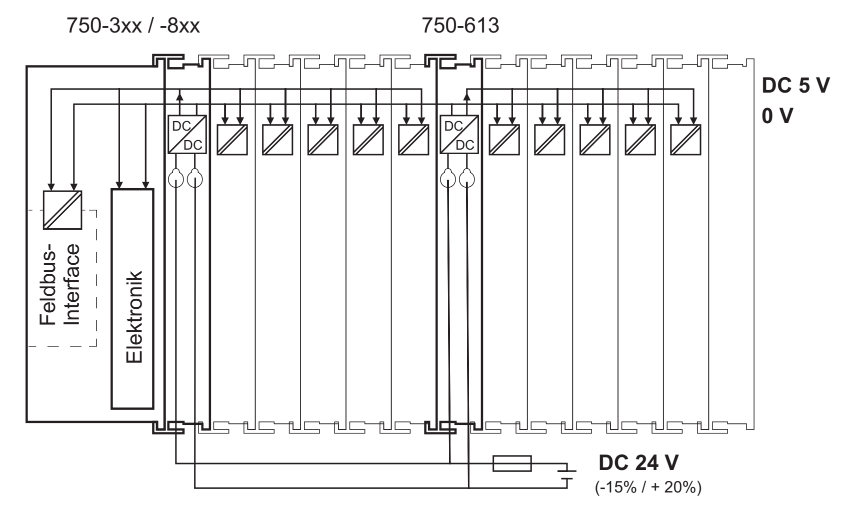


Abbildung 5: Systemspannung für Standard-Feldbuskoppler/-controller und erweiterter ECO-Feldbuskoppler

Hinweis



Rücksetzen des Systems nur gleichzeitig bei allen Versorgungsmodulen!
Führen Sie das Rücksetzen des Systems durch Aus- und Einschalten der Systemversorgung gleichzeitig bei allen Versorgungsmodulen (Feldbuskoppler/-controller und Potentialeinspeiseklemme mit Busnetzteil 750-613) durch.

3.6.2.2 Auslegung

Hinweis



Empfehlung
Eine stabile Netzversorgung kann nicht immer und überall vorausgesetzt werden. Sie sollten daher geregelte Netzteile verwenden, um die Qualität der Versorgungsspannung zu gewährleisten.

Die Versorgungskapazität der Feldbuskoppler/-controller bzw. der Potentialeinspeiseklemme mit Busnetzteil (750-613) kann den technischen Daten der Komponenten entnommen werden.

Tabelle 3: Auslegung

Interne Stromaufnahme *)	Stromaufnahme über Systemspannung. 5 V für Elektronik der Busklemmen und Feldbuskoppler/-controller
Summenstrom für Busklemmen *)	Verfügbarer Strom für die Busklemmen. Wird vom Busnetzteil bereitgestellt. Siehe Feldbuskoppler/-controller und Potentialeinspeiseklemme mit Busnetzteil (750-613)

*) vgl. aktuellen Katalog, Handbücher, Internet



Beispiel:

Berechnung Stromaufnahme am PROFIBUS DP/FMS Feldbuskoppler 750-301

Interne Stromaufnahme	350 mA bei 5 V
Summenstrom für Busklemmen	1650 mA bei 5 V
Summe $I_{(5\ V)\ ges}$	**2000 mA bei 5V**

Für jede Busklemme ist die interne Stromaufnahme in den technischen Daten der Busklemme angegeben. Um den Gesamtbedarf zu ermitteln, werden die Werte aller Busklemmen im Knoten summiert.

Hinweis



Summenstrom für Busklemmen beachten, evtl. Potential neu einspeisen!
Sobald die Summe der internen Stromaufnahmen den Summenstrom für Busklemmen übersteigt, müssen Sie eine Potentialeinspeiseklemme mit Busnetzteil (750-613) setzen. Platzieren Sie diese vor die Position, an der der zulässige Summenstrom die Grenze überschreiten würde.

Beispiel:

Berechnung Summenstrom an einem Standard-Feldbuskoppler/-controller

In einem Knoten mit dem PROFIBUS DP/FMS Feldbuskoppler 750-301 sollen eingesetzt werden:
20 Relaisklemmen (750-517) und 30 Digitaleingangsklemmen (750-405)

Interne Stromaufnahme	20 * 90 mA =	1800 mA
	30 * 2 mA =	60 mA
Summe		**1860 mA**

Der PROFIBUS DP/FMS Feldbuskoppler 750-301 kann 1650 mA für die Busklemmen bereitstellen (siehe Datenblatt). Folglich muss eine Potentialeinspeiseklemme mit Busnetzteil (750-613), z. B. in der Mitte des Knotens, vorgesehen werden.

Hinweis





Empfehlung
Sie können mit der WAGO-ProServe®-Software **smartDESIGNER** den Aufbau eines Feldbusknotens konfigurieren. Über die integrierte Plausibilitätsprüfung können Sie die Konfiguration überprüfen.

Der maximale Eingangsstrom der 24-V-Systemversorgung beträgt 500 mA.
Die genaue Stromaufnahme ($I_{(24\ V)}$) kann mit folgenden Formeln ermittelt werden:

Koppler oder Controller

$I_{(5\ V)\ ges.}$ = Summe aller Stromaufnahmen der angereihten Busklemmen + interne Stromaufnahme des Kopplers/Controllers

Potentialeinspeiseklemme 750-613

$I_{(5\ V)\ ges.}$ = Summe aller Stromaufnahmen der angereihten Busklemmen an der Potentialeinspeiseklemme

$$\text{Eingangsstrom } I_{(24\ V)} = \frac{5\ V}{24\ V} * \frac{I_{(5\ V)\ ges.}}{\eta}$$

η = 0.87 (87 % Netzteilwirkungsgrad bei Nennlast 24 V)

Hinweis



Bei Test der Stromaufnahme alle Ausgänge aktivieren!
Übersteigt die Stromaufnahme der Einspeisestelle für die 24-V-Systemversorgung 500 mA, kann die Ursache ein falsch ausgelegter Knoten oder ein Defekt sein.
Sie müssen bei dem Test alle Ausgänge, insbesondere die Relaisklemme, aktivieren.



3.6.3 Feldversorgung

3.6.3.1 Anschluss

Sensoren und Aktoren können direkt in 1-/4-Leiteranschlusstechnik an den jeweiligen Kanal der Busklemmen angeschlossen werden. Die Versorgung der Sensoren und Aktoren übernimmt die Busklemme. Die Ein- und Ausgangstreiber einiger Busklemmen benötigen die feldseitige Versorgungsspannung.

Die feldseitige Versorgungsspannung wird am Feldbuskoppler/-controller (DC 24 V) eingespeist. In diesem Fall handelt es sich um eine passive Einspeisung ohne Schutzeinrichtung. Zur Einspeisung anderer Feldpotentiale, z. B. AC 230 V, stehen Einspeiseklemmen zur Verfügung.

Ebenso können mit Hilfe der Einspeiseklemmen unterschiedliche Potentialgruppen aufgebaut werden. Die Anschlüsse sind paarweise mit einem Leistungskontakt verbunden.

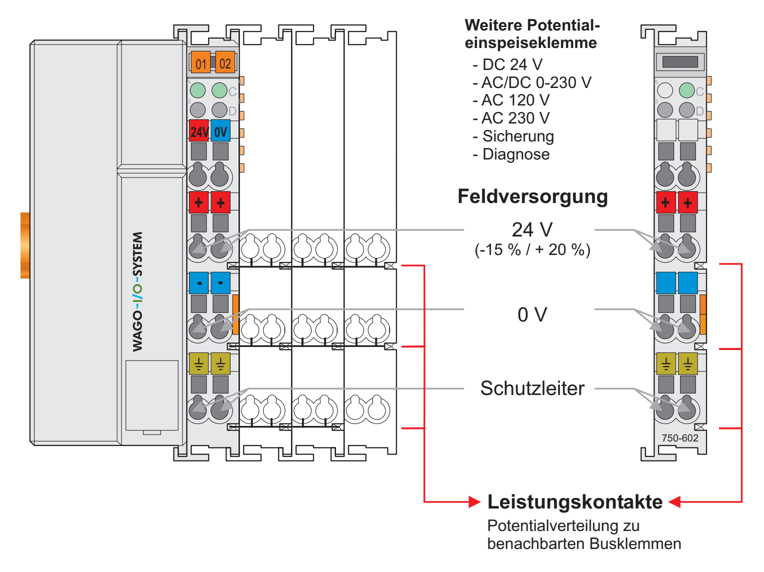


Abbildung 6: Feldversorgung (Sensor/Aktor) für Standard-Feldbuskoppler/-controller und erweiterte ECO-Feldbuskoppler

Die Weiterleitung der Versorgungsspannung für die Feldseite erfolgt automatisch durch Anrasten der jeweiligen Busklemme über die Leistungskontakte.

Die Strombelastung der Leistungskontakte darf 10 A nicht dauerhaft überschreiten. Die Strombelastbarkeit zwischen zwei Anschlussklemmen ist mit der Belastbarkeit der Verbindungsdrähte identisch.

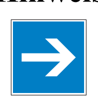
Durch Setzen einer zusätzlichen Einspeiseklemme wird die über die Leistungskontakte geführte Feldversorgung unterbrochen. Ab dort erfolgt eine neue Einspeisung, die auch einen Potentialwechsel beinhalten kann.

Hinweis



Potential bei Unterbrechung der Leistungskontakte neu einspeisen!
Einige Busklemmen besitzen keine oder nur einzelne Leistungskontakte (abhängig von der E/A-Funktion). Dadurch wird die Weitergabe des entsprechenden Potentials unterbrochen. Wenn bei nachfolgenden Busklemmen eine Feldversorgung erforderlich ist, müssen Sie eine Potentialeinspeiseklemme einsetzen.
Beachten Sie die Datenblätter der einzelnen Busklemmen.

Hinweis



Bei unterschiedlichen Potentialgruppen Distanzklemme verwenden!
Bei einem Knotenaufbau mit verschiedenen Potentialgruppen, z. B. der Wechsel von DC 24 V auf AC 230 V, sollten Sie eine Distanzklemme einsetzen. Die optische Trennung der Potentiale mahnt zur Vorsicht bei Verdrahtungs- und Wartungsarbeiten. Somit können Sie die Folgen von Verdrahtungsfehlern vermeiden.



3.6.3.2 Absicherung

Die interne Absicherung der Feldversorgung ist für verschiedene Feldspannungen über entsprechende Potentialeinspeiseklemmen möglich.

Tabelle 4: Potentialeinspeiseklemmen

Artikelnummer	Feldspannung
750-601	24 V DC, Einspeisung/Sicherung
750-609	230 V AC, Einspeisung/Sicherung
750-615	120 V AC, Einspeisung/Sicherung
750-610	24 V DC, Einspeisung/Sicherung/Diagnose
750-611	230 V AC, Einspeisung/Sicherung/Diagnose

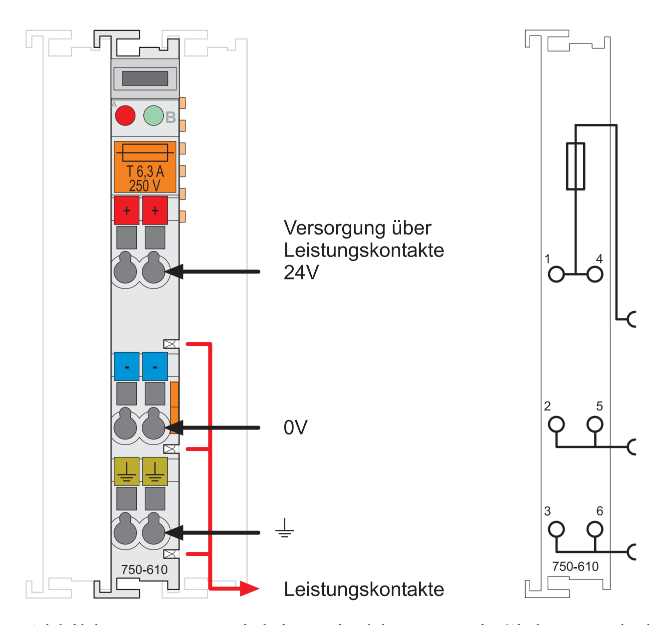


Abbildung 7: Potentialeinspeiseklemme mit Sicherungshalter (Beispiel 750-610)

ACHTUNG



Auf max. Verlustleistung und ggf. UL-Zulassung achten!
Bei Einspeiseklemmen mit Sicherungshalter dürfen Sie nur Sicherungen mit einer max. Verlustleitung von 1,6 W (IEC 127) einsetzen. Bei Anlagen, die eine UL-Zulassung besitzen, achten Sie zusätzlich darauf, dass Sie nur UL-zugelassene Sicherungen verwenden.

Um eine Sicherung einzulegen, zu wechseln oder um nachfolgende Busklemmen spannungsfrei zu schalten, kann der Sicherungshalter herausgezogen werden. Dazu wird, z. B. mit einem Schraubendreher, in einen der beidseitig vorhandenen Schlitze gegriffen und der Halter herausgezogen.



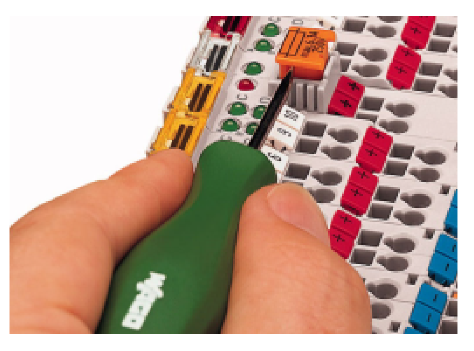


Abbildung 8: Sicherungshalter ziehen

Der Sicherungshalter wird geöffnet, indem die Abdeckung zur Seite geklappt wird.

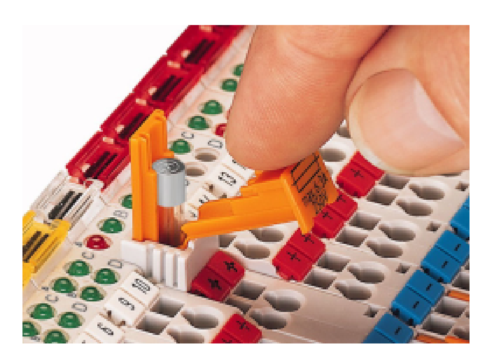


Abbildung 9: Sicherungshalter öffnen

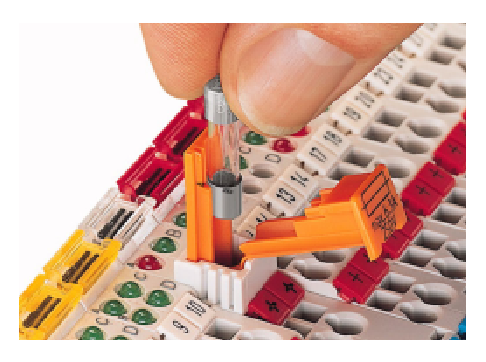


Abbildung 10: Sicherung wechseln

Nach dem Sicherungswechsel wird der Sicherungshalter in seine ursprüngliche Position zurückgeschoben.

Alternativ kann die Absicherung extern erfolgen. Hierbei bieten sich die Sicherungsklemmen der WAGO-Serien 281 und 282 an.

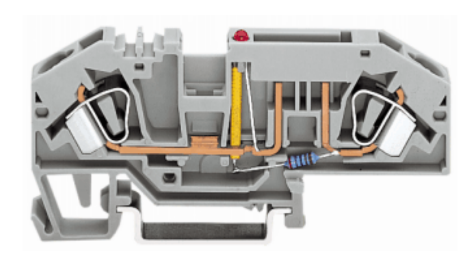


Abbildung 11: Sicherungsklemmen für Kfz-Sicherungen, Serie 282

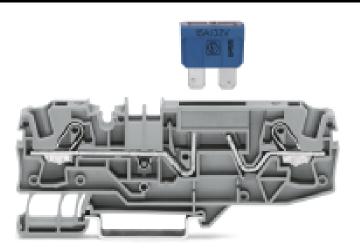


Abbildung 12: Sicherungsklemmen für Kfz-Sicherungen, Serie 2006

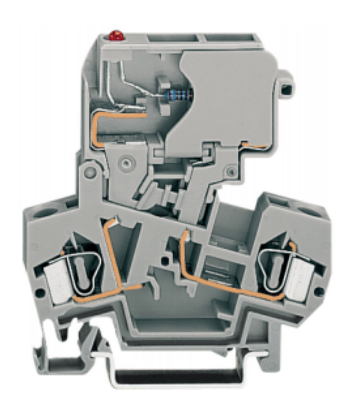


Abbildung 13: Sicherungsklemmen mit schwenkbarem Sicherungshalter, Serie 281

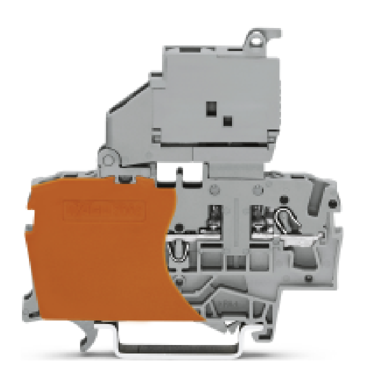


Abbildung 14: Sicherungsklemmen mit schwenkbarem Sicherungshalter, Serie 2002

3.6.4 Ergänzende Einspeisevorschriften

Das WAGO-I/O-SYSTEM 750 kann auch im Schiffbau bzw. Off-/Onshore-Bereichen (z. B. Arbeitsplattformen, Verladeanlagen) eingesetzt werden. Dies wird durch die Einhaltung der Anforderungen einflussreicher Klassifikationsgesellschaften, z. B. Germanischer Lloyd und Lloyds Register, nachgewiesen.

Der zertifizierte Betrieb des Systems erfordert Filterklemmen für die 24V-Versorgung.

Tabelle 5: Filterklemmen für die 24-V-Versorgung

Artikel-Nr.	Bezeichnung	Beschreibung
750-626	Supply Filter	Filterklemme für Systemversorgung und Feldversorgung (24 V, 0 V), d. h. für Feldbuskoppler/-Controller und Bus Einspeisung (750-613)
750-624	Supply Filter	Filterklemme für die 24V-Feldversorgung (750-602, 750-601, 750-610)

Daher ist zwingend folgendes Einspeisekonzept zu beachten.

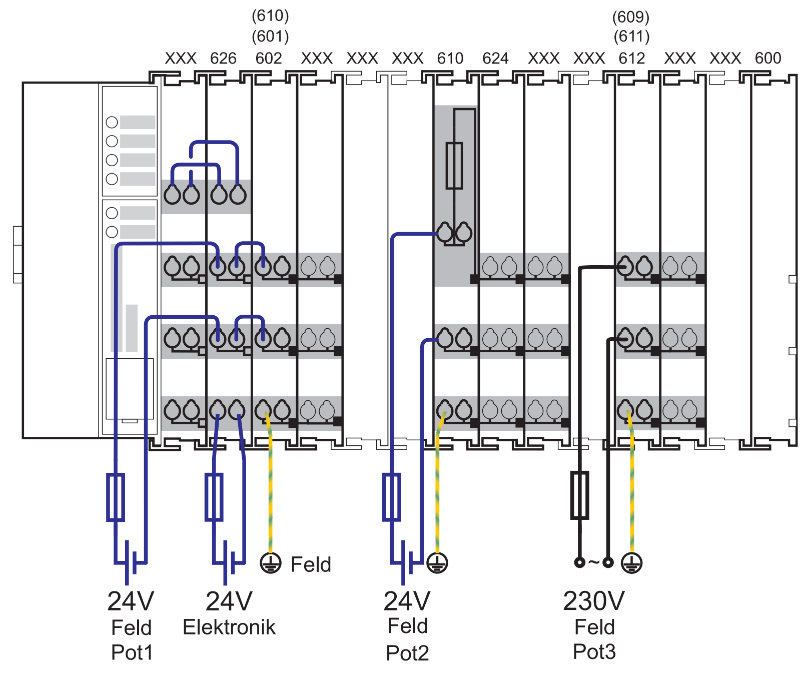


Abbildung 15: Einspeisekonzept

Hinweis



Zusätzliche Potentialeinspeiseklemme als Schutzleiter/zur Absicherung
Setzen Sie hinter der Filterklemme 750-626 eine zusätzliche Potentialeinspeiseklemme 750-601/602/610 dann ein, wenn Sie den Schutzleiter auf dem unteren Leistungskontakt benötigen oder wenn Sie eine Absicherung wünschen.



3.6.5 Versorgungsbeispiel

Hinweis



System- und Feldversorgung getrennt einspeisen!
Speisen Sie die Systemversorgung und die Feldversorgung getrennt ein, um bei aktorseitigen Kurzschlüssen den Busbetrieb zu gewährleisten.

L1
L2
L3
N
PE

Versorgung - System
230V ~ 24V —

Versorgung - Feld
230V ~ 24V —

Versorgung - Feld
10 A

Schirmung
Erdungssammelleiter

1) Distanzklemme empfohlen
2) Ringspeisung empfohlen

a) Potentialeinspeisung am Koppler / Controller über externe Einspeiseklemme
b) Potentialeinspeisung mit Busnetzteil
c) Potentialeinspeisung passiv
d) Potentialeinspeisung mit Sicherungshalter/ Diagnose

Abbildung 16: Versorgungsbeispiel für Standard-Feldbuskoppler/-controller und erweiterte ECO-Feldbuskoppler



3.6.6 Netzgeräte

Das WAGO-I/O-SYSTEM 750 benötigt zum Betrieb eine 24V-Gleichspannung (Systemversorgung) mit einer maximalen Abweichung von -15 % bzw. +20 %.

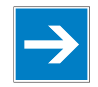
Hinweis



Empfehlung
Eine stabile Netzversorgung kann nicht immer und überall vorausgesetzt werden. Daher sollten Sie geregelte Netzteile verwenden, um die Qualität der Versorgungsspannung zu gewährleisten.

Für kurze Spannungseinbrüche ist ein Puffer (200 µF pro 1 A Laststrom) einzuplanen.

Hinweis



Netzausfallzeit nicht nach IEC61131-2!
Beachten Sie, dass die Netzausfallzeit von 10 ms nach IEC61131-2 in einem Maximalausbau nicht eingehalten wird.

Je Einspeisestelle für die Feldversorgung ist der Strombedarf individuell zu ermitteln. Dabei sind alle Lasten durch Feldgeräte und Busklemmen zu berücksichtigen. Die Feldversorgung hat ebenfalls Einfluss auf die Busklemmen, da die Ein- und Ausgangstreiber einiger Busklemmen die Spannung der Feldversorgung benötigen.

Hinweis



System- und Feldversorgung getrennt einspeisen!
Speisen Sie die Systemversorgung und die Feldversorgung getrennt ein, um bei aktorseitigen Kurzschlüssen den Busbetrieb zu gewährleisten.

Tabelle 6: WAGO-Netzgeräte

WAGO-Netzgeräte Artikelnummer	Beschreibung
787-612	Primär getaktet; DC 24 V; 2,5 A Eingangsspannung AC 230 V
787-622	Primär getaktet, DC 24 V; 5 A Eingangsspannung AC 230 V
787-632	Primär getaktet; DC 24 V; 10 A Eingangsspannungsbereich AC 230/115 V
	Schienenmontierbare Netzgeräte auf Universal Montagesockel
288-809	AC 115 V/DC 24 V; 0,5 A
288-810	AC 230 V/DC 24 V; 0,5 A
288-812	AC 230 V/DC 24 V; 2 A
288-813	AC 115 V/DC 24 V; 2 A



3.7 Erdung

3.7.1 Erdung der Tragschiene

3.7.1.1 Rahmenaufbau

Beim Rahmenaufbau ist die Tragschiene mit dem elektrisch leitenden Schrankrahmen bzw. Gehäuse verschraubt. Der Rahmen bzw. das Gehäuse muss geerdet sein. Über die Verschraubung wird auch die elektrische Verbindung hergestellt. Somit ist die Tragschiene geerdet.

GEFAHR



Auf ausreichende Erdung achten!
Achten Sie auf eine einwandfreie elektrische Verbindung zwischen der Tragschiene und dem Rahmen bzw. Gehäuse, um eine ausreichende Erdung sicher zu stellen.

3.7.1.2 Isolierter Aufbau

Ein isolierter Aufbau liegt dann vor, wenn es konstruktiv keine direkte leitende Verbindung zwischen Schrankrahmen oder Maschinenteilen und der Tragschiene gibt. Hier muss über einen elektrischen Leiter entsprechend geltender nationaler Sicherheitsvorschriften die Erdung aufgebaut werden.

Hinweis



Empfehlung
Der optimale Aufbau ist eine metallische Montageplatte mit Erdungsanschluss, die elektrisch leitend mit der Tragschiene verbunden ist.

Die separate Erdung der Tragschiene kann einfach mit Hilfe der WAGO-Schutzleiterklemmen aufgebaut werden.

Tabelle 7: WAGO-Schutzleiterklemmen

Artikelnummer	Beschreibung
283-609	1-Leiter-Schutzleiterklemme kontaktiert den Schutzleiter direkt auf der Tragschiene; Anschlussquerschnitt: 0,2 -16 mm^2 **Hinweis:** Abschlussplatte (283-320) mitbestellen

3.7.2 Funktionserde

Die Funktionserde erhöht die Störunempfindlichkeit gegenüber elektromagnetischen Einflüssen. Einige Komponenten des I/O-Systems besitzen einen Tragschienenkontakt, der elektromagnetische Störungen zur Tragschiene ableitet.

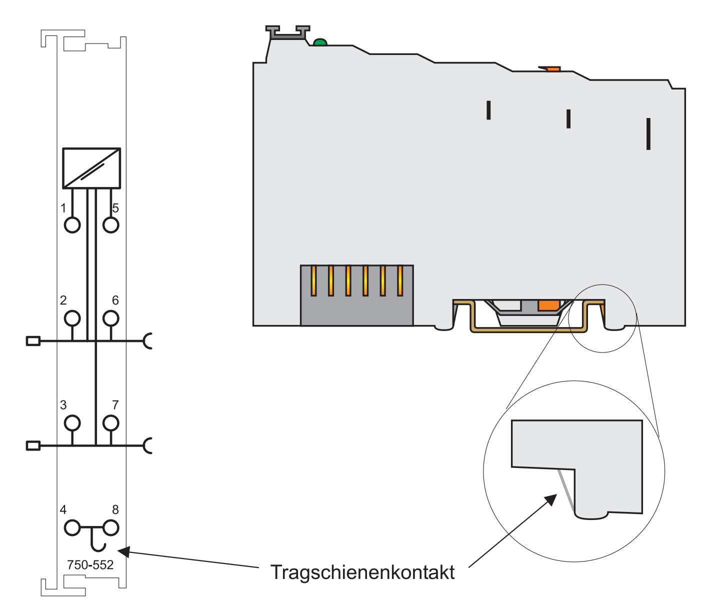


Abbildung 17: Tragschienenkontakt

GEFAHR

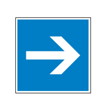


Auf ausreichende Erdung achten!
Achten Sie auf den einwandfreien Kontakt zwischen dem Tragschienenkontakt und der Tragschiene. Die Tragschiene muss geerdet sein.
Beachten Sie dazu die Tragschieneneigenschaften, siehe Kapitel „Montage auf Tragschiene > Tragschieneneigenschaften“.

3.7.3 Schutzerde

Für die Feldebene wird die Schutzerde an den unteren Anschlussklemmen der Einspeiseklemmen aufgelegt und über die unteren Leistungskontakte an die benachbarten Busklemmen weitergereicht. Besitzt die Busklemme den unteren Leistungskontakt, kann der Schutzleiteranschluss der Feldgeräte direkt an die unteren Anschlussklemmen der Busklemme angeschlossen werden.

Hinweis



Potential bei Unterbrechung der Leistungskontakte neu einspeisen!
Ist die Verbindung der Leistungskontakte für den Schutzleiter innerhalb des Knotens unterbrochen, z. B. durch eine 4-Kanal-Busklemme, müssen Sie das Potential neu einspeisen.

Eine Ringspeisung des Erdpotentials kann die Systemsicherheit erhöhen. Für den Fall, dass eine Busklemme aus der Potentialgruppe gezogen wird, bleibt das Erdpotential erhalten.

Bei der Ringspeisung wird der Schutzleiter am Anfang und am Ende einer Potentialgruppe angeschlossen.

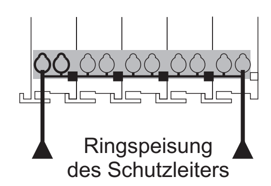


Abbildung 18: Ringspeisung

Hinweis



Vorschriften zur Verwendung von Schutzerde beachten!
Halten Sie die jeweils örtlichen und national gültigen Vorschriften zur Instandhaltung und Überprüfung der Schutzerde ein.

3.8 Schirmung

3.8.1 Allgemein

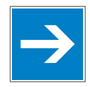
Die Schirmung der Daten- und Signalleitungen verringert die elektromagnetischen Einflüsse und erhöht damit die Signalqualität. Messfehler, Datenübertragungsfehler und sogar Zerstörung durch Überspannung werden vermieden.

Hinweis



Schirmung durchgehend, großflächig und am Einlass verlegen!
Eine durchgängige Schirmung ist zwingend erforderlich, um die technischen Angaben bezüglich der Messgenauigkeit zu gewährleisten.
Legen Sie die Schirmung der Kabel großflächig auf das Erdpotential. Damit können eingestreute Störungen leicht abfließen.
Sie sollten die Schirmung schon am Einlass des Schrankes bzw. Gehäuses auflegen, um Störungen schon am Einlass abzufangen.

Hinweis



Starkstrom führende Kabel separat verlegen!
Verlegen Sie Daten- und Signalleitungen separat von allen Starkstrom führenden Kabeln.

3.8.2 Busleitungen

Die Schirmung der Busleitung ist in den jeweiligen Aufbaurichtlinien und Normen des Bussystems beschrieben.

3.8.3 Signalleitungen

Die Busklemmen für Analogsignale sowie einige Schnittstellen-Busklemmen besitzen Anschlussklemmen für den Schirm.

Hinweis



Verbessern der Schirmung durch großflächige Auflage!
Eine verbesserte Schirmung wird erreicht, wenn Sie den Schirm vorher großflächig auflegen. Hier empfiehlt sich, dass Sie z. B. das WAGO-Schirm-Anschlusssystem einsetzen. Dies empfiehlt sich insbesondere bei Anlagen mit großer Ausdehnung, bei denen nicht ausgeschlossen werden kann, dass Ausgleichsströme fließen oder hohe impulsförmige Ströme, z. B. ausgelöst durch atmosphärische Entladung, auftreten können.



3.8.4 WAGO-Schirm-Anschlusssystem

Das WAGO-Schirm-Anschlusssystem besteht aus Schirm-Klemmbügeln, Sammelschienen und diversen Montagefüßen, um eine Vielzahl von Aufbauten zu realisieren. Siehe Katalog W4 Band 3 Kapitel 10.

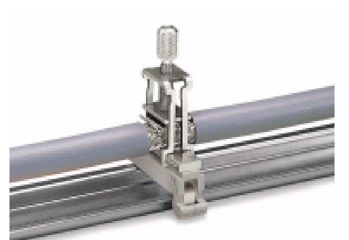
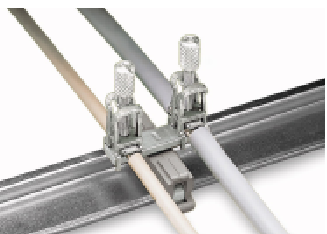
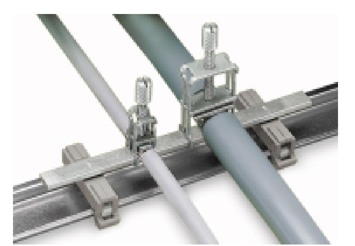




Abbildung 19: Beispiel WAGO-Schirm-Anschlusssystem

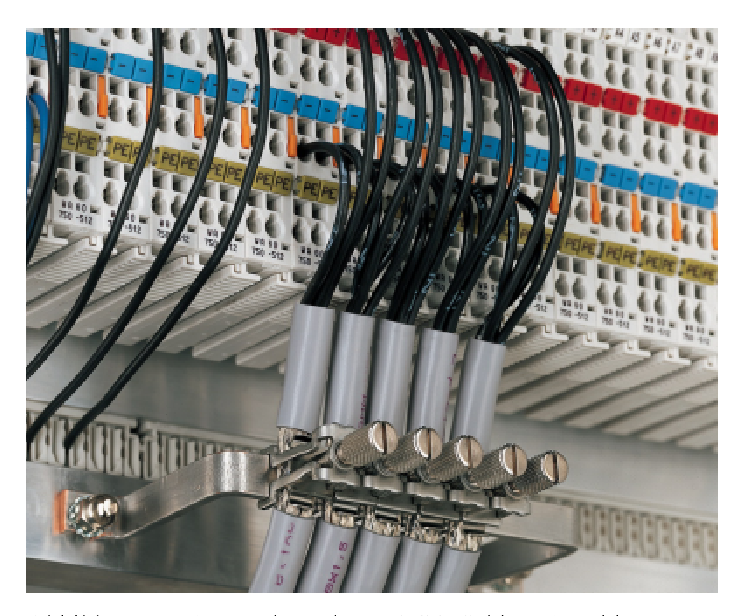


Abbildung 20: Anwendung des WAGO-Schirm-Anschlusssystems

4 Gerätebeschreibung

Der programmierbare Feldbuscontroller 750-841 (kurz: PFC) kombiniert die Funktionalität eines Feldbuskopplers zur Anschaltung an den Feldbus ETHERNET mit der einer Speicherprogrammierbaren Steuerung (SPS).

In dem Feldbuscontroller werden sämtliche Eingangssignale der Sensoren zusammengeführt. Nach Anschluss des Feldbuscontrollers ermittelt der Feldbuscontroller alle in dem Knoten gesteckten Busklemmen und erstellt daraus ein lokales Prozessabbild. Hierbei kann es sich um eine gemischte Anordnung von analogen (Datenaustausch wortweise) und digitalen (Datenaustausch bitweise) Klemmen handeln.

Das lokale Prozessabbild wird in einen Eingangs- und Ausgangsdatenbereich unterteilt.

Die Daten der analogen Busklemmen werden in der Reihenfolge ihrer Position nach dem Feldbuscontroller in das Prozessabbild gemappt.

Die Bits der digitalen Busklemmen werden zu Worten zusammengefügt und im Anschluss an die analogen ebenfalls in das Prozessabbild gemappt. Ist die Anzahl der digitalen E/As größer als 16 Bit, beginnt der Feldbuscontroller automatisch ein weiteres Wort.

Entsprechend der IEC 61131-3-Programmierung erfolgt die Bearbeitung der Prozessdaten vor Ort in dem PFC. Die daraus erzeugten Verknüpfungsergebnisse können direkt an die Aktoren ausgegeben oder über den Bus an die übergeordnete Steuerung übertragen werden.

Wahlweise kann der Feldbuscontroller über die ETHERNET-Medientypen „100BaseTX“ oder „10BaseT“ mit übergeordneten Systemen kommunizieren. Hierzu nutzt der Feldbuscontroller die RJ-45-Buchse.

Die Erstellung des Applikationsprogramms erfolgt mit WAGO-I/O-*PRO* CAA gemäß IEC 61131-3, wobei die Basis von WAGO-I/O-*PRO* CAA das Standard Programmiersystem CoDeSys der Firma 3S ist, das mit den Target-Dateien für alle WAGO-Feldbuscontroller spezifisch erweitert wurde.

Für die IEC 61131-3 Programmierung stellt der Feldbuscontroller 512 KB Programmspeicher, 256 KB Datenspeicher und 24 KB Retain-Speicher zur Verfügung.

Der Anwender hat Zugriff auf alle Feldbus- und E/A-Daten.

Um Prozessdaten via ETHERNET zu versenden, unterstützt der Feldbuscontroller eine Reihe von Netzwerkprotokollen.

Für den Prozessdatenaustausch sind das MODBUS/TCP (UDP)-Protokoll und das ETHERNET/IP-Protokoll implementiert.
Beide Kommunikationsprotokolle können wahlweise oder parallel verwendet



werden. Dazu wird die Schreibberechtigung auf die Busklemmen, d. h. der Zugriff auf diese von dem PFC aus über MODBUS/TCP oder EtherNet/IP, in einer XML-Datei festgelegt.

Als Konfigurations- und Diagnoseprotokolle stehen BootP, HTTP, DHCP, DNS, SNTP, FTP, SNMP und SMTP zur Verfügung.

Der Anwender kann Clients und Server über eine interne Socket-API für alle Transportprotokolle (TCP, UDP, u.s.w.) mit Funktionsbausteinen programmieren. Zur Funktionserweiterung sind Library-Funktionen verfügbar.

Mit der IEC 61131-3 Bibliothek „SysLibRTC.lib“ wird beispielsweise eine gepufferte Echtzeituhr mit Datum, Zeit (Auflösung 1 Sekunde), Alarmfunktionen und einem Timer eingebunden. Diese Uhr wird bei einem Energieausfall durch eine Hilfsenergie überbrückt.

Der Feldbuscontroller basiert auf einer 32-Bit-CPU und ist multitasking-fähig, d. h. mehrere Programme können quasi-gleichzeitig ausgeführt werden.

Für die Konfiguration und Verwaltung des Systems bietet der Feldbuscontroller ein internes Dateisystem sowie einen integrierten Webserver.

Informationen über die Konfiguration und den Status des Feldbusknotens sind als HTML-Seiten in dem Feldbuscontroller gespeichert und können über einen Webbrowser ausgelesen werden. Darüber hinaus lassen sich über das implementierte Dateisystem auch eigene HTML-Seiten hinterlegen oder Programme direkt aufrufen.

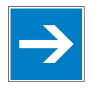
Tabelle 8: Kompatibilität

Programmiertool:	WAGO-I/O-*PRO* 32 759-332		WAGO-I/O-*PRO* CAA 750-333				
-Version	V2.1	V2.2.6	V2.3.2.5	V2.3.2.7	V2.3.3.4	V2.3.3.6	V2.3.4.3
Feldbuscontroller:							
750-841	-	-	✓	✓	✓	SW ≥ 06	SW ≥ 09

Erläuterung:

-	Feldbuscontroller arbeitet NICHT mit der WAGO-I/O-*PRO*-Version.
✓	Feldbuscontroller arbeitet mit der WAGO-I/O-*PRO*-Version, unabhängig von der Softwareversion des Feldbuscontrollers.
SW ≥ xy	Feldbuscontroller arbeitet mit der WAGO-I/O-*PRO*-Version, wenn der Feldbuscontroller die Softwareversion xy oder höher hat.

Hinweis



Versionsstände für die Komptibilität beachten!
Beachten Sie, dass die CoDeSys Netzwerkvariablen ab WAGO-I/O-*PRO* V2.3.3.6 und höher von den Feldbuscontrollern 750-841 mit der Software SW ≥ 06 unterstützt werden.
Die WEB-Visualisierung ab WAGO-I/O-*PRO* V2.3.4.3 und höher werden von den Feldbuscontrollern 750-841 mit der Software SW ≥ 09.unterstützt..



4.1 Ansicht

Die Ansicht zeigt drei Einheiten:

- Auf der linken Seite befindet sich der Feldbusanschluss.
- In dem mittleren Bereich sind LEDs zur Statusanzeige des Betriebes, zur Buskommunikation, zur Fehlermeldung und Diagnose sowie die Service-Schnittstelle zu finden.
- Die rechte Seite der Ansicht zeigt die Geräteeinspeisung mit Netzteil zur Systemversorgung und zur Feldversorgung der angereihte Busklemmen über Leistungskontakte.
  LEDs zeigen die Betriebsspannung für das System und die Leistungskontakte an.

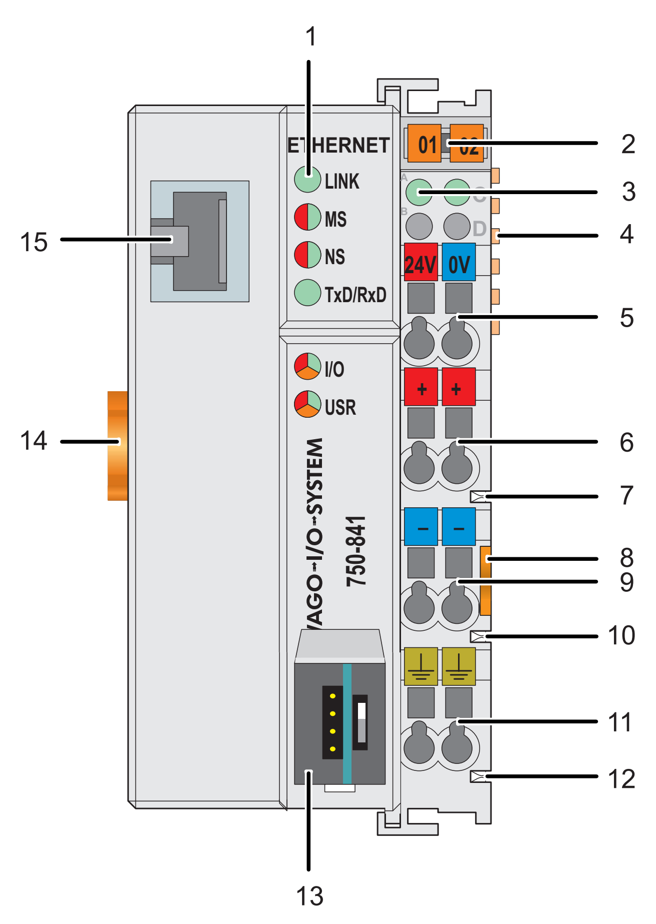


Abbildung 21: Ansicht ETHERNET TCP/IP-Feldbuscontroller



Tabelle 9: Legende zur Ansicht ETHERNET TCP/IP-Feldbuscontroller

Nr.	Be-zeichnung	Bedeutung	Details siehe Kapitel:
1	LINK, MS, NS, TxD/RxD, I/O, USR	Status-LEDs Feldbus	„Gerätebeschreibung“ > „Anzeigeelemente“
2	---	Gruppenbezeichnungsträger (herausziehbar) mit zusätzlicher Beschriftungsmöglichkeit auf zwei Mini-WSB-Schildern	---
3	A, B bzw. C	Status-LEDs System-/Feldversorgung	„Gerätebeschreibung“ > „Anzeigeelemente“
4	---	Datenkontakte	„Geräte anschließen“ > „Klemmenbus/Daten-kontakte“
5	24 V, 0 V	CAGE CLAMP®–Anschlüsse Systemversorgung	„Geräte anschließen“ > „Leiter an CAGE CLAMP® anschließen“
6	+	CAGE CLAMP®–Anschlüsse Feldversorgung 24 V DC	„Geräte anschließen“ > „Leiter an CAGE CLAMP® anschließen“
7	---	Leistungskontakt 24 V DC	„Geräte anschließen“ > „Leistungskontakte/ Feldversorgung“
8	---	Entriegelungslasche	„Montieren“ > „Geräte einfügen und entfernen“
9	-	CAGE CLAMP®–Anschlüsse Feldversorgung 0 V	„Systembeschreibung“ > „Spannungsversorgung“
10	---	Leistungskontakt 0 V	„Geräte anschließen“ > „Leistungskontakte/ Feldversorgung“
11	(Erdung)	CAGE CLAMP®–Anschlüsse Feldversorgung (Erdung)	„Systembeschreibung“ > „Spannungsversorgung“
12	---	Leistungskontakt (Erdung)	„Geräte anschließen“ > „Leistungskontakte/ Feldversorgung“
13	---	Service-Schnittstelle (Klappe geöffnet)	„Gerätebeschreibung“ > „Bedienelemente“
14	---	Verriegelungsscheibe	„Montieren“ > „Geräte einfügen und entfernen“
15	---	Feldbusanschluss RJ-45	„Gerätebeschreibung“ > „Anschlüsse“

4.2 Anschlüsse

4.2.1 Geräteeinspeisung

Die Versorgung wird über Klemmen mit CAGE CLAMP®-Anschluss eingespeist.

Das integrierte Netzteil erzeugt die erforderlichen Spannungen zur Versorgung der Elektronik und der angereihten Busklemmen.

Das Feldbus-Interface ist mittels Übertrager galvanisch von dem elektrischen Potential des Gerätes getrennt.

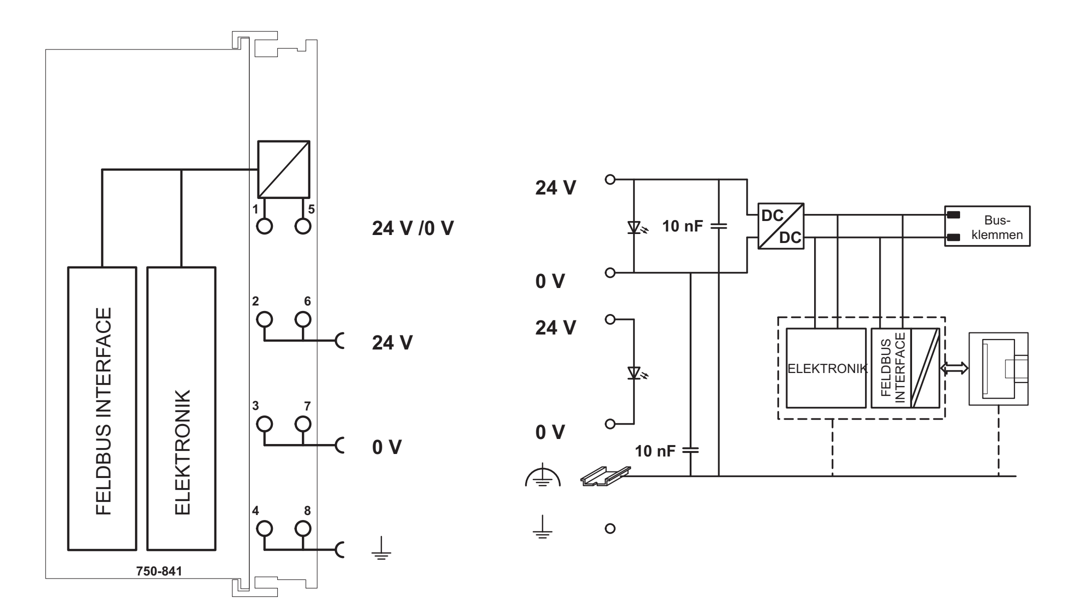


Abbildung 22: Geräteeinspeisung

4.2.2 Feldbusanschluss

Der Anschluss an den Feldbus erfolgt über einen RJ-45-Steckverbinder, auch „Westernstecker“ genannt.
Die Beschaltung der RJ-45-Buchse ist entsprechend den Vorgaben für 100BaseTX.
Als Verbindungsleitung wird vom ETHERNET-Standard ein Twisted-Pair-Kabel der Kategorie 5e vorgeschrieben. Dabei können Leitungen des Typs S-UTP (Screened-Unshielded Twisted Pair) sowie STP (Shielded Twisted Pair) mit einer maximalen Segmentlänge von 100 m benutzt werden.
Die Anschlussstelle ist so konzipiert, dass nach Steckeranschluss ein Einbau in einen 80 mm hohen Schaltkasten möglich ist.

Tabelle 10: Busanschluss und Steckerbelegung, RJ-45-Stecker

Ansicht	Kontakt	Signal	
	1	TD +	Transmit Data +
	2	TD -	Transmit Data -
	3	RD +	Receive Data +
	4		nicht belegt
	5		nicht belegt
	6	RD -	Receive Data -
	7		nicht belegt
Abbildung 23: RJ-45-Stecker	8		nicht belegt

ACHTUNG



Nicht in Telekommunikationsnetzen einsetzen!
Verwenden Sie Geräte mit ETHERNET-/RJ-45-Anschluss ausschließlich in LANs. Verbinden Sie diese Geräte niemals mit Telekommunikationsnetzen.

4.3 Anzeigeelemente

Der Betriebszustand des Feldbuscontrollers bzw. des Knoten wird über Leuchtmelder in Form von Leuchtdioden (LEDs) signalisiert. Diese sind zum Teil mehrfarbig (rot, grün oder rot/grün (=orange)) ausgeführt.

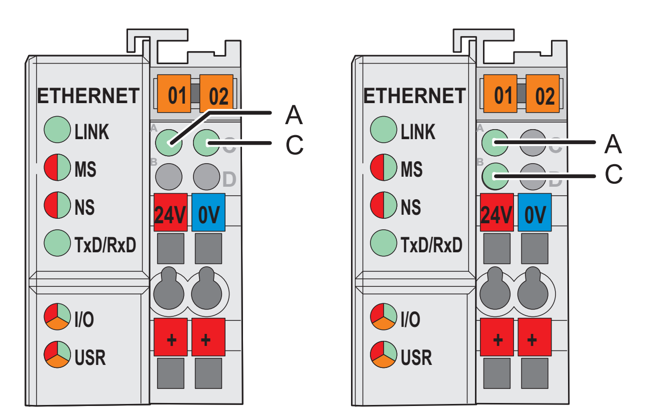


Abbildung 24: Anzeigeelemente (zwei Fertigungsvarianten)

Zur Diagnose der verschiedenen Bereiche für Feldbus, Knoten und Versorgungsspannung werden entsprechend drei Gruppen von LEDs unterschieden:

Tabelle 11: Anzeigeelemente Feldbusstatus

LED	Farbe	Bedeutung
LINK	grün	zeigt eine Verbindung zum physikalischen Netzwerk an
MS	rot/grün	zeigt den Knoten-Status an
NS	rot/grün	zeigt den Netzwerk-Status an
TxD/RxD	rot/grün/orange	zeigt das Stattfinden eines Datenaustausches an

Tabelle 12: Anzeigeelemente Knotenstatus

LED	Farbe	Bedeutung
I/O	rot/grün/orange	zeigt den Betrieb des Knotens an und signalisiert mittels eines Blinkcodes auftretende Fehler
USR	rot/grün/orange	zeigt, angesteuert aus dem Anwenderprogramm entsprechend der Visualisierungsprogrammierung, Informationen zu Klemmenbus-fehlern an

Tabelle 13: Anzeigeelemente Versorgungsspannungsstatus

LED	Farbe	Bedeutung
A	grün	zeigt den Status der Betriebsspannung des Systems an
B oder C	grün	zeigt den Status der Betriebsspannung der Leistungskontakte an (diese LED kann sich fertigungsabhängig entweder an Position B oder C befinden)

Information



Weitere Informationen zu der LED-Signalisierung
Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose“ >„LED-Signalisierung“.



4.4 Bedienelemente

4.4.1 Service-Schnittstelle

Die Service-Schnittstelle befindet sich hinter der Abdeckklappe.

Sie wird für die Kommunikation mit WAGO-I/O-*CHECK*, WAGO-I/O-*PRO* CAA und zum Firmware-Download genutzt.

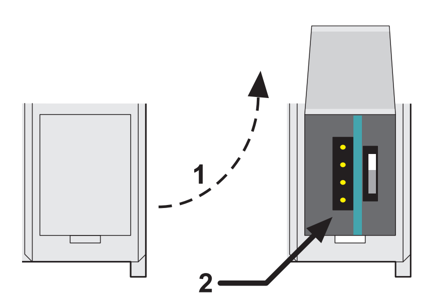


Abbildung 25: Service-Schnittstelle zur Konfiguration und Programmierung (geschlossene und geöffnete Klappe)

Tabelle 14: Service-Schnittstelle

Nummer	Beschreibung
1	Klappe geöffnet
2	Konfigurations- und Programmierschnittstelle

ACHTUNG



Gerät muss spannungsfrei sein!
Um Geräteschäden zu vermeiden, ziehen und stecken Sie das Kommunikationskabel nur, wenn das Gerät spannungsfrei ist!

An die 4-polige Stiftleiste wird das Kommunikationskabel mit der Art.-Nr. 750-920 angeschlossen.



4.4.2 Betriebsartenschalter

Der Betriebsartenschalter befindet sich hinter der Abdeckklappe.

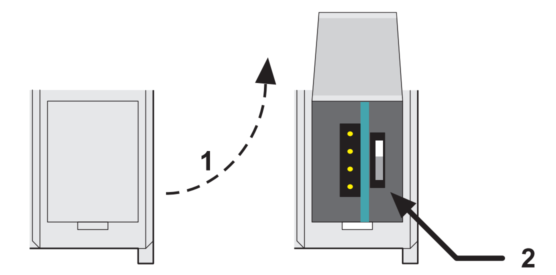


Abbildung 26: Betriebsartenschalter (geschlossene und geöffnete Klappe der Service-Schnittstelle)

Tabelle 15: Service-Schnittstelle

Nummer	Beschreibung
1	Klappe geöffnet
2	Betriebsartenschalter

Der Betriebsartenschalter bestimmt das Laden, Starten und Stoppen der SPS-Applikation durch den Controller. Bei diesem Druck-/Schiebeschalter gibt es 3 Rast-Stellungen und eine Tastfunktion.
Der Schiebeschalter ist für eine Betätigungshäufigkeit nach EN 61131T2 ausgelegt.

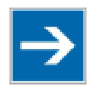
ACHTUNG



Sachschäden durch gesetzte Ausgänge!
Beachten Sie, dass gesetzte Ausgänge weiterhin gesetzt bleiben, wenn Sie im laufenden Betrieb den Betriebsartenschalter von „RUN“ auf „STOP“ schalten! Da das Programm dann nicht mehr bearbeitet wird, sind software-seitige Abschaltungen z. B. durch Initiatoren, unwirksam.
Programmieren bzw. definieren Sie deshalb alle Ausgänge, damit diese bei einem Programmstopp in einen sicheren Zustand schalten.



Hinweis



Definieren der Ausgänge für einen Programm-Stopp!
Um die Ausgänge bei Programm-Stopp in einen sicheren Zustand zu schalten, definieren Sie den Status der Ausgänge bei „STOP“.

1. Öffnen Sie dazu in dem Web-based Management-System (WBM) über den Link "PLC" eine Web-Seite, auf der Sie die Funktion ***Process image - Set outputs to zero, if user program is stopped***, festlegen können.
2. Aktivieren Sie durch Setzen eines Häkchens in das Kontrollkästchen diese Funktion, dann werden alle Ausgänge auf Null gesetzt, ist diese Funktion nicht aktiviert, verbleiben die Ausgänge auf dem letzten aktuellen Wert.

Hinweis



Bei Software-Start/-Stopp Betriebsartenschalterstellung unerheblich!
Die Stellung des Betriebsartenschalters ist für das Starten und Stoppen der PFC-Applikation aus WAGO-I/O-*PRO* heraus unerheblich.

Je nachdem, in welcher der drei statischen Positionen „Oben“, „Mitte“ oder „Unten“ sich der Schalter bei einem PowerOn oder einem Hard- oder Software-Reset befindet, ist eine der folgenden Funktionen aktiv:

Tabelle 16: Betriebsartenschalterstellungen, statische Positionen bei PowerOn/Reset

Stellung des Betriebsartenschalters	**Funktion**
Position Oben	"RUN" - Programmbearbeitung aktivieren, Boot-Projekt (wenn vorhanden) wird gestartet
Position Mitte	"STOP" - Programmbearbeitung stoppen, PFC-Applikation wird angehalten
Position Unten	Der WAGO Firmwareloader wird nach einem PowerOn/Reset auf der Service-Schnittstelle gestartet.

Wird während des laufenden Betriebs ein Stellungswechsel des Schalters vorgenommen, führt der Controller die folgenden Funktionen aus:

Tabelle 17: Betriebsartenschalterstellungen, dynamische Positionen im laufenden Betrieb

Stellungswechsel des Betriebsartenschalters	**Funktion**
Von obere in mittlere Position	"STOP" - Programmbearbeitung stoppen, PFC-Applikation wird angehalten.
Von mittlere in obere Position	"RUN" - Programmbearbeitung aktivieren, Boot-Projekt (wenn vorhanden) wird gestartet.
Von mittlere in untere Position	Es erfolgt keine Reaktion. Nach PowerOn/Reset wird der Bootstraploader gestartet.
Von untere in mittlere Position	Es erfolgt keine Reaktion.
Niederdrücken (z. B. mit Schraubendreher)	Hardware-Reset. Alle Ausgänge werden rückgesetzt; Variablen werden auf 0 bzw. auf FALSE oder auf einen Initialwert gesetzt. Retain-Variable, bzw. Merker werden nicht verändert. Der Hardware-Reset kann sowohl bei STOP als auch bei RUN in jeder Stellung des Betriebsartenschalters ausgeführt werden! Neuanlauf des Feldbuscontrollers.

Der Wechsel der Betriebsart erfolgt intern am Ende eines PFC-Zyklus.



4.5 Technische Daten

4.5.1 Gerätedaten

Tabelle 18: Technische Daten – Gerätedaten

Breite	51 mm
Höhe (ab Oberkante Tragschiene)	65 mm
Tiefe	100 mm
Gewicht	184 g
Schutzart	IP 20

4.5.2 Systemdaten

Tabelle 19: Technische Daten – Systemdaten

Max. Anzahl Busteilnehmer	Limitiert durch ETHERNET-Spezifikation
Übertragungsmedium	Twisted Pair S/UTP 100 Ω Cat 5
Busanschluss	RJ-45
Bussegmentlänge $_{max}$	100 m zwischen Hub und Feldbuscontroller
Netzwerklänge $_{max}$	Netzwerklänge durch ETHERNET Spezifikation limitiert
Übertragungsrate	10/100 Mbit/s
Protokolle	MODBUS/TCP (UDP), ETHERNET/IP, HTTP, BootP, DHCP, DNS, SNTP, FTP, SNMP
Programmierung	WAGO-I/O-*PRO* CAA
IEC-61131-3	AWL, KOP, FUP, ST, AS
Max. Anzahl Socket-Verbindungen	3 HTTP, 15 MODBUS/TCP, 10 FTP, 2 SNMP, 5 für IEC-61131-3-Programme, 2 für WAGO-I/O-*PRO* CAA, 128 für EtherNet/IP
Anzahl Busklemmen - mit Busverlängerung	64 250
Programmspeicher	512 kByte
Datenspeicher	256 kByte
Remanentspeicher	24 kByte (16 k retain, 8 k Merker)

4.5.3 Versorgung

Tabelle 20: Technische Daten – Versorgung

Spannungsversorgung	DC 24 V (-25 % ... +30 %)
Eingangstrom $_{max.}$	500 mA bei 24 V
Netzteilwirkungsgrad	87 %
Interne Stromaufnahme	300 mA bei 5 V
Summenstrom für Busklemmen	1700 mA bei 5 V
Potentialtrennung	500 V System/Versorgung
Spannung über Leistungskontakte	DC 24 V (-25 % ... +30 %)
Strom über Leistungskontakte $_{max.}$	DC 10 A

4.5.4 Feldbus MODBUS/TCP

Tabelle 21: Technische Daten – Feldbus MODBUS/TCP

Eingangsprozessabbild $_{max}$	2 kByte
Ausgangsprozessabbild $_{max}$	2 kByte
Eingangsvariablen $_{max}$	512 Byte
Ausgangsvariablen $_{max}$	512 Byte

4.5.5 Zubehör

Tabelle 22: Technische Daten – Zubehör

Mini-WSB-Schnellbezeichnungssystem	
WAGO-I/O-*PRO* CAA	

4.5.6 Anschlusstechnik

Tabelle 23: Technische Daten Anschlusstechnik

Anschlusstechnik	CAGE CLAMP®
Leiterquerschnitt	0,08 mm² ... 2,5 mm², AWG 28-14
Abisolierlänge	8 ... 9 mm / 0.33 in
Leistungskontakte	Messer-/Federkontakt, selbstreinigend
Spannungsabfall bei I $_{max.}$	< 1 V bei 64 Busklemmen
Datenkontakte	Gleitkontakte, 1,5 µm hartvergoldet, selbstreinigend



4.5.7 Klimatische Umweltbedingungen

Tabelle 24: Technische Daten – Klimatische Umweltbedingungen

Betriebstemperaturbereich	0 °C ... 55 °C
Lagertemperaturbereich	-25 °C ... +85 °C
Relative Feuchte (ohne Betauung)	max. 95 %
Beanspruchung durch Schadstoffe	gem. IEC 60068-2-42 u. IEC 60068-2-43
Max. Schadstoffkonzentration bei einer relativen Feuchte <75%	$SO_2 \leq 25$ ppm $H_2S \leq 10$ ppm
Besondere Bedingungen	Die Komponenten dürfen nicht ohne Zusatzmaßnahmen an Orten eingesetzt werden, an denen Staub, ätzende Dämpfe, Gase oder ionisierte Strahlung auftreten können.

4.5.8 Mechanische Belastbarkeit gem. IEC 61131-2

Tabelle 25: Technische Daten - Mechanische Belastbarkeit gem. IEC 61131-2

Prüfung	Frequenzbereich	Grenzwert
IEC 60068-2-6 Vibration	5 Hz ≤ f < 9 Hz	1,75 mm Amplitude (dauerhaft) 3,5 mm Amplitude (kurzzeitig)
	9 Hz ≤ f < 150 Hz	0,5 g (dauerhaft) 1 g (kurzzeitig)
	Anmerkung zur Vibrationsprüfung: a) Frequenzänderung: max. 1 Oktave/Minute b) Vibrationsrichtung: 3 Achsen	
IEC 60068-2-27 Stoß		15 g
	Anmerkung zur Stoßprüfung: a) Art des Stoßes: Halbsinus b) Stoßdauer: 11 ms c) Stoßrichtung: je 3 Stöße in positive und negative Richtung der 3 senkrecht zueinanderstehenden Achsen des Prüflings	
IEC 60068-2-32 Freier Fall	1 m (Gerät in Originalverpackung)	

*) QP: Quasi Peak

4.6 Zulassungen

Information



Weitere Informationen zu Zulassungen
Detaillierte Hinweise zu den Zulassungen können Sie dem Dokument „Übersicht Zulassungen **WAGO-I/O-SYSTEM 750**“ entnehmen. Dieses finden Sie auf der DVD „AUTOMATION Tools and Docs“ (Art.-Nr.: 0888-0412) oder im Internet unter: www.wago.com → Service → Dokumentation → WAGO-I/O-SYSTEM 750 → Systembeschreibung.

Folgende Zulassungen wurden für den Feldbuskoppler/-controller 750-841 erteilt:

CE Konformitätskennzeichnung


$_{C}UL_{US}$ (UL508)

Folgende Ex-Zulassungen wurden für den Feldbuskoppler/-controller 750-841 erteilt:

Ex	TÜV	07 ATEX 554086 X	I M2 Ex d I II 3 G Ex nA IIC T4 II 3 D Ex tD A22 IP6X T135°C
	Erlaubte Betriebstemperatur:		0 °C ≤ T_A ≤ +60 °C
Ex	TÜV	TUN 09.0001X	Ex d I Ex nA IIC T4 Ex tD A22 IP6X T135°C
	Erlaubte Betriebstemperatur:		0 °C ≤ T_A ≤ +60 °C
Ex	$_{C}UL_{US}$	ANSI/ISA 12.12.01	Class I, Div2 ABCD T4

Folgende Schiffszulassungen wurden für den Feldbuskoppler/-controller 750-841 erteilt:

	ABS (American Bureau of Shipping)	
	BSH (Bundesamt für Seeschifffahrt und Hydrographie)	
	BV (Bureau Veritas)	
	DNV (Det Norske Veritas)	Class B
	GL (Germanischer Lloyd)	Cat. A, B, C, D (EMC 1)
	KR (Korean Register of Shipping)	
	LR (Lloyd's Register)	Env. 1, 2, 3, 4
	NKK (Nippon Kaiji Kyokai)	
	PRS (Polski Rejestr Statków)	
	RINA (Registro Italiano Navale)	

Information



Weitere Information zu den Schiffszulassungen
Beachten Sie zu den Schiffszulassungen das Kapitel „Ergänzende Einspeisevorschriften".

4.7 Normen und Richtlinien

Der Feldbuskoppler/-controller 750-841 erfüllt folgende EMV-Normen:

EMV CE-Störfestigkeit	gem. EN 61000-6-2: 2005
EMV CE-Störaussendung	gem. EN 61000-6-3: 2007
EMV Schiffbau-Störfestigkeit	gem. Germanischer Lloyd (2003)
EMV Schiffbau-Störaussendung	gem. Germanischer Lloyd (2003)

Der Feldbuskoppler/-controller 750-841 erfüllt die Anforderungen an Störaussendung im Wohnbereich.

5 Montieren

5.1 Einbaulage

Neben dem horizontalen und vertikalen Einbau sind alle anderen Einbaulagen erlaubt.

Hinweis



Bei vertikalem Einbau Endklammer verwenden!
Montieren Sie beim vertikalen Einbau zusätzlich unterhalb des Feldbusknotens eine Endklammer, um den Feldbusknoten gegen Abrutschen zu sichern.

WAGO Artikel 249-116	Endklammer für TS 35, 6 mm breit
WAGO Artikel 249-117	Endklammer für TS 35, 10 mm breit

5.2 Gesamtaufbau

Die nutzbare Länge der Busklemmen hinter dem Feldbuskoppler/-controller beträgt 780 mm inklusiv Endklemme. Die Breite der Endklemme beträgt 12 mm. Die übrigen Busklemmen verteilen sich also auf einer Länge von maximal 768 mm.

Beispiele:

- An einen Feldbuskoppler/-controller können 64 Ein- und Ausgangsklemmen der Breite 12 mm gesteckt werden.
- An einen Feldbuskoppler/-controller können 32 Ein- und Ausgangsklemmen der Breite 24 mm gesteckt werden.

Ausnahme:

Die Anzahl der gesteckten Busklemmen hängt außerdem von dem jeweiligen Feldbuskoppler/-controller ab, an dem sie betrieben werden. Beispielsweise beträgt die maximale Anzahl der Busklemmen an einem PROFIBUS-Feldbuskoppler/-controller 63 ohne Endklemme.

ACHTUNG



Maximale Gesamtausdehnung eines Knotens beachten!
Die maximale Gesamtausdehnung eines Knotens ohne Feldbuskoppler/-controller darf eine Länge von 780 mm nicht überschreiten.
Beachten Sie zudem Einschränkungen einzelner Feldbuskoppler/-controller.

Hinweis



Gesamtausdehnung mit WAGO-Klemmenbusverlängerung erhöhen!
Mit der WAGO-Klemmenbusverlängerung können Sie die Gesamtausdehnung eines Knotens erhöhen. Bei einem solchen Aufbau stecken Sie nach der letzten Klemme eines Knotens eine Klemmenbusverlängerungsendklemme 750-627. Diese verbinden Sie per RJ-45-Kabel mit der Klemmenbusverlängerungskopplerklemme 750-628 eines weiteren Klemmenblocks.
An einer Klemmenbusverlängerungsendklemme 750-627 können Sie bis zu 10 Klemmenbusverlängerungskopplerklemmen 750-628 anschließen. Damit können Sie einen Feldbusknoten in maximal 11 Blöcke aufteilen.
Die zulässige Kabellänge zwischen zwei Blöcken beträgt 5 Meter (für weitere Informationen, siehe Handbuch der Klemmen 750-627/-628). Die zulässige Gesamtkabellänge in einem Feldbusknoten beträgt 70 Meter.



5.3 Montage auf Tragschiene

5.3.1 Tragschieneneigenschaften

Alle Komponenten des Systems können direkt auf eine Tragschiene gemäß EN 50022 (TS 35, DIN Rail 35) aufgerastet werden.

ACHTUNG



Ohne Freigabe keine WAGO-fremden Tragschienen verwenden!
WAGO liefert normkonforme Tragschienen, die optimal für den Einsatz mit dem WAGO-I/O-SYSTEM geeignet sind. Sollten Sie andere Tragschienen einsetzen, muss eine technische Untersuchung und eine Freigabe durch WAGO Kontakttechnik GmbH & Co. KG vorgenommen werden.

Tragschienen weisen unterschiedliche mechanische und elektrische Merkmale auf. Für den optimalen Aufbau des Systems auf einer Tragschiene sind Randbedingungen zu beachten:

- Das Material muss korrosionsbeständig sein.
- Die meisten Komponenten besitzen zur Ableitung von elektromagnetischen Einflüssen einen Ableitkontakt zur Tragschiene. Um Korrosionseinflüsse vorzubeugen, darf dieser verzinnte Tragschienenkontakt mit dem Material der Tragschiene kein galvanisches Element bilden, das eine Differenzspannung über 0,5 V (Kochsalzlösung von 0,3 % bei 20 °C) erzeugt.
- Die Tragschiene muss die im System integrierten EMV-Maßnahmen und die Schirmung über die Busklemmenanschlüsse optimal unterstützen.
- Eine ausreichend stabile Tragschiene ist auszuwählen und ggf. mehrere Montagepunkte (alle 20 cm) für die Tragschiene zu nutzen, um Durchbiegen und Verdrehung (Torsion) zu verhindern.
- Die Geometrie der Tragschiene darf nicht verändert werden, um den sicheren Halt der Komponenten sicherzustellen. Insbesondere beim Kürzen und Montieren darf die Tragschiene nicht gequetscht oder gebogen werden.
- Der Rastfuß der Komponenten reicht in das Profil der Tragschiene hinein. Bei Tragschienen mit einer Höhe von 7,5 mm sind Montagepunkte (Verschraubungen) unter dem Knoten in der Tragschiene zu versenken (Senkkopfschrauben oder Blindnieten).
- Die Metallfedern auf der Gehäuseunterseite müssen niederimpedanten Kontakt zur Tragschiene haben (möglichst breitflächige Auflage).

5.3.2 WAGO-Tragschienen

Die WAGO-Tragschienen erfüllen die elektrischen und mechanischen Anforderungen.

Tabelle 26: WAGO-Tragschienen

Artikelnummer	Beschreibung
210-113 /-112	35 x 7,5; 1 mm Stahl gelb chromatiert; gelocht/ungelocht
210-114 /-197	35 x 15; 1,5 mm Stahl gelb chromatiert; gelocht/ungelocht
210-118	35 x 15; 2,3 mm Stahl gelb chromatiert; ungelocht
210-198	35 x 15; 2,3 mm Kupfer; ungelocht
210-196	35 x 7,5; 1 mm Alu; ungelocht

5.4 Abstände

Für den gesamten Feldbusknoten sind Abstände zu benachbarten Komponenten, Kabelkanälen und Gehäuse-/Rahmenwänden einzuhalten.

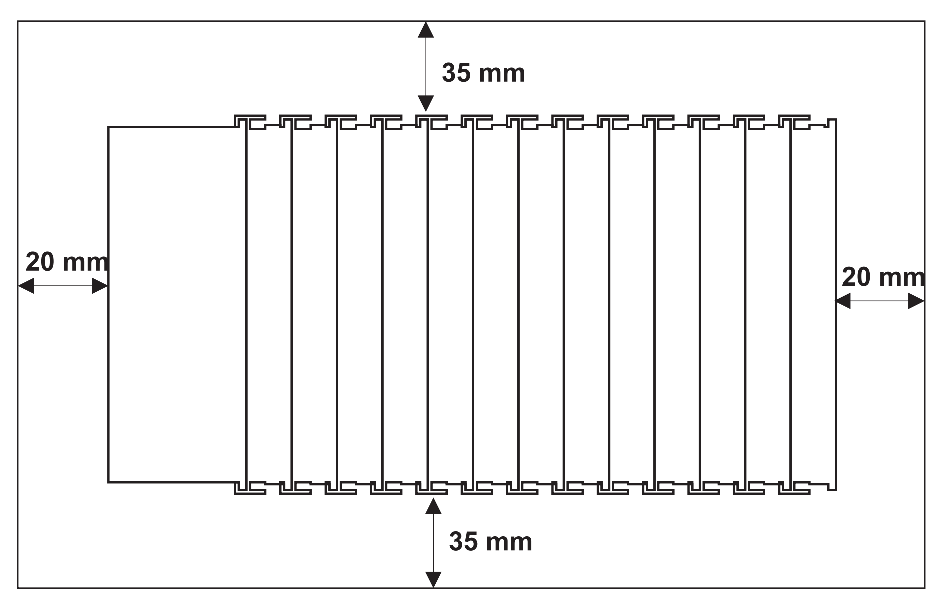


Abbildung 27: Abstände

Die Abstände schaffen Raum zur Wärmeableitung und Montage bzw. Verdrahtung. Ebenso verhindern die Abstände zu Kabelkanälen, dass leitungsgebundene elektromagnetische Störungen den Betrieb beeinflussen.



5.5 Montagereihenfolge

Feldbuskoppler/-controller und Busklemmen des **WAGO-I/O-SYSTEM 750** werden direkt auf eine Tragschiene gemäß EN 50022 (TS 35) aufgerastet.

Die sichere Positionierung und Verbindung erfolgt über ein Nut- und Feder-System. Eine automatische Verriegelung garantiert den sicheren Halt auf der Tragschiene.

Beginnend mit dem Feldbuskoppler/-controller werden die Busklemmen entsprechend der Projektierung aneinandergereiht. Fehler bei der Projektierung des Knotens bezüglich der Potentialgruppen (Verbindungen über die Leistungskontakte) werden erkannt, da Busklemmen mit Leistungskontakten (Messerkontakte) nicht an Busklemmen angereiht werden können, die weniger Leistungskontakte besitzen.

VORSICHT



Verletzungsgefahr durch scharfkantige Messerkontakte!
Da die Messerkontakte sehr scharfkantig sind, besteht bei unvorsichtiger Hantierung mit den Busklemmen Verletzungsgefahr.

ACHTUNG



Busklemmen in vorgegebener Reihenfolge stecken!
Stecken Sie die Busklemmen nie aus Richtung der Endklemme. Ein Schutzleiter-Leistungskontakt, der in eine Busklemme ohne Kontakt, z. B. eine Digitaleingangsklemme mit 4 Kanälen, eingeschoben wird, besitzt eine verringerte Luft- und Kriechstrecke zu dem benachbarten Kontakt.

ACHTUNG



Aneinanderreihen von Busklemmen nur bei offener Nut!
Einige Busklemmen besitzen keine oder nur wenige Leistungskontakte. Das Aneinanderreihen einiger Busklemmen ist deshalb mechanisch nicht möglich, da die Nuten für die Messerkontakte oben geschlossen sind.

Hinweis



Busabschluss nicht vergessen!
Stecken Sie immer eine Bus-Endklemme 750-600 an das Ende des Feldbusknotens! Die Bus-Endklemme muss in allen Feldbusknoten mit Feldbuskopplern/-controllern des WAGO-I/O-SYSTEM 750 eingesetzt werden, um eine ordnungsgemäße Datenübertragung zu garantieren!

5.6 Geräte einfügen und entfernen

GEFAHR



Vorsicht bei der Unterbrechung von PE!
Stellen Sie sicher, dass durch das Entfernen einer Busklemme und der damit verbundenen Unterbrechung von PE kein Zustand eintreten kann, der zur Gefährdung von Menschen oder Geräten führen kann.
Sehen Sie zur Vermeidung von Unterbrechungen eine Ringspeisung des Schutzleiters vor, siehe Kapitel „Erdung/Schutzleiter“ im Handbuch „Systembeschreibung WAGO-I/O-SYSTEM 750“.

ACHTUNG



Arbeiten an Geräten nur spannungsfrei durchführen!
Arbeiten unter Spannung können zu Schäden an den Geräten führen. Schalten Sie daher die Spannungsversorgung ab, bevor Sie an den Geräten arbeiten.



5.6.1 Feldbuskoppler/-controller einfügen

1. Wenn Sie den Feldbuskoppler/-controller gegen einen bereits vorhandenen Feldbuskoppler/-controller austauschen, positionieren Sie den neuen Feldbuskoppler/-controller so, dass die Nut- und Federverbindungen zur nachfolgenden Busklemme im Eingriff sind.

2. Rasten Sie den Feldbuskoppler/-controller auf die Tragschiene auf.

3. Drehen Sie die Verriegelungsscheibe mit einer Schraubendreherklinge, bis die Nase der Verriegelungsscheibe hinter der Tragschiene einrastet (siehe nachfolgende Abbildung). Damit ist der Feldbuskoppler/-controller auf der Tragschiene gegen Verkanten gesichert.

Mit dem Einrasten des Feldbuskopplers/-controllers sind die elektrischen Verbindungen der Datenkontakte und (soweit vorhanden) der Leistungskontakte zur gegebenenfalls nachfolgenden Busklemme hergestellt.

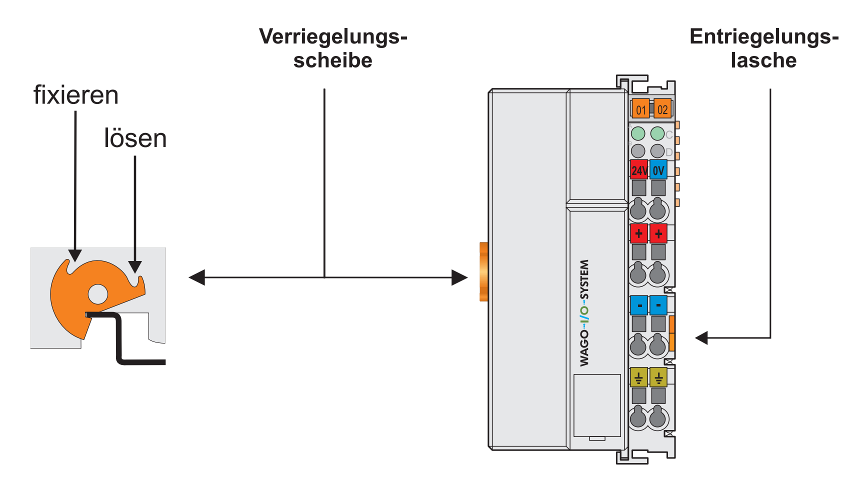


Abbildung 28: Verriegelung Standard-Feldbuskoppler/-controller

5.6.2 Feldbuskoppler/-controller entfernen

1. Drehen Sie die Verriegelungsscheibe mit einer Schraubendreherklinge, bis die Nase der Verriegelungsscheibe nicht mehr hinter der Tragschiene eingerastet ist.

2. Ziehen Sie die Feldbuskoppler/-controller an der Entriegelungslasche aus dem Verband.

Mit dem Herausziehen des Feldbuskopplers/-controllers sind die elektrischen Verbindungen der Datenkontakte bzw. Leistungskontakte zu nachfolgenden Busklemmen wieder getrennt.

5.6.3 Busklemme einfügen

1. Positionieren Sie die Busklemme so, dass die Nut- und Federverbindungen zum Feldbuskoppler/-controller oder zur vorhergehenden und gegebenenfalls zur nachfolgenden Busklemme im Eingriff sind.

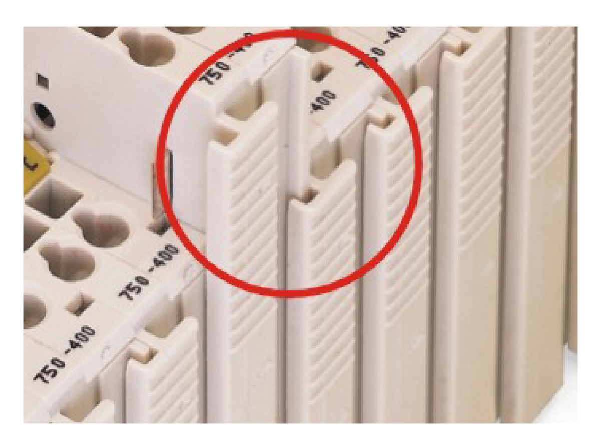


Abbildung 29: Busklemme einsetzen

2. Drücken Sie die Busklemme in den Verband bis die Busklemme auf der Tragschiene einrastet.

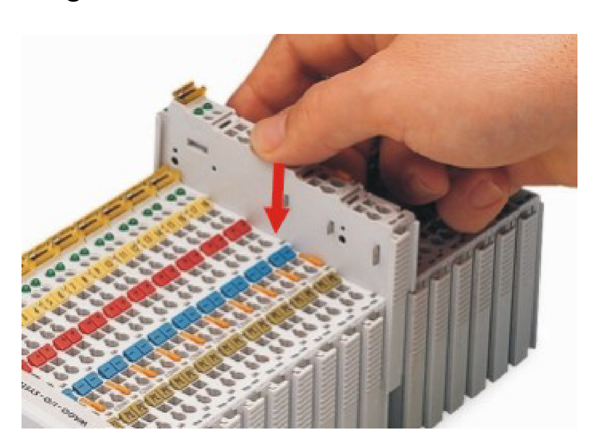


Abbildung 30: Busklemme einrasten

Mit dem Einrasten der Busklemme sind die elektrischen Verbindungen der Datenkontakte und (soweit vorhanden) der Leistungskontakte zum Feldbuskoppler/-controller oder zur vorhergehenden und gegebenenfalls zur nachfolgenden Busklemme hergestellt.



5.6.4 Busklemme entfernen

1. Ziehen Sie die Busklemme an der Entriegelungslasche aus dem Verband.

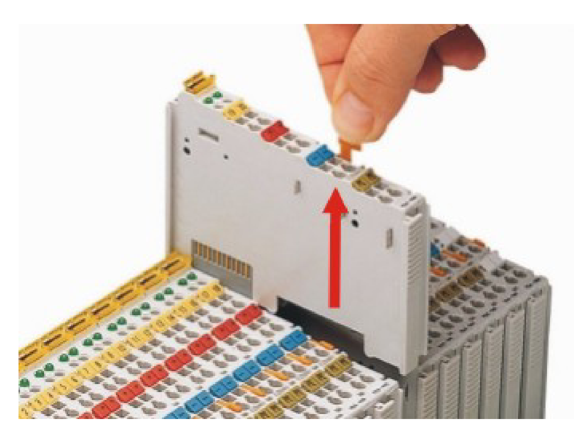


Abbildung 31: Busklemme lösen

Mit dem Herausziehen der Busklemme sind die elektrischen Verbindungen der Datenkontakte bzw. Leistungskontakte wieder getrennt.

6 Geräte anschließen

6.1 Datenkontakte/Klemmenbus

Die Kommunikationen zwischen Feldbuskoppler/-controller und Busklemmen sowie die Systemversorgung der Busklemmen erfolgt über den Klemmenbus. Er besteht aus 6 Datenkontakten, die als selbstreinigende Goldfederkontakte ausgeführt sind.

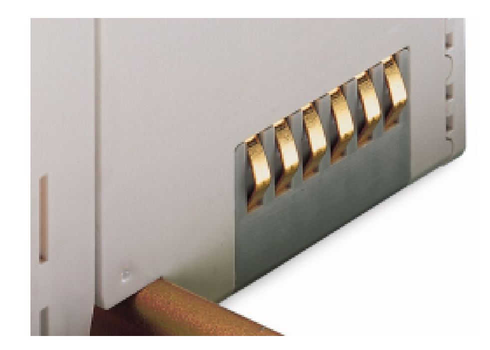


Abbildung 32: Datenkontakte

ACHTUNG



Busklemmen nicht auf Goldfederkontakte legen!
Um Verschmutzung und Kratzer zu vermeiden, legen Sie die Busklemmen nicht auf die Goldfederkontakte.

ESD



Auf gute Erdung der Umgebung achten!
Die Geräte sind mit elektronischen Bauelementen bestückt, die bei elektrostatischer Entladung zerstört werden können. Achten Sie beim Umgang mit den Geräten auf gute Erdung der Umgebung (Personen, Arbeitsplatz und Verpackung). Berühren Sie keine elektrisch leitenden Bauteile, z. B. Datenkontakte.



6.2 Leistungskontakte/Feldversorgung

VORSICHT



Verletzungsgefahr durch scharfkantige Messerkontakte!
Da die Messerkontakte sehr scharfkantig sind, besteht bei unvorsichtiger Hantierung mit den Busklemmen Verletzungsgefahr.

Auf der rechten Seite aller Feldbuskoppler/-controller und einiger Busklemmen befinden sich selbstreinigende Leistungskontakte. Die Leistungskontakte leiten die Versorgungsspannung für die Feldseite weiter. Die Kontakte sind berührungssicher als Federkontakte ausgeführt.
Als Gegenstück sind auf der linken Seite der Busklemmen entsprechende Messerkontakte vorhanden.

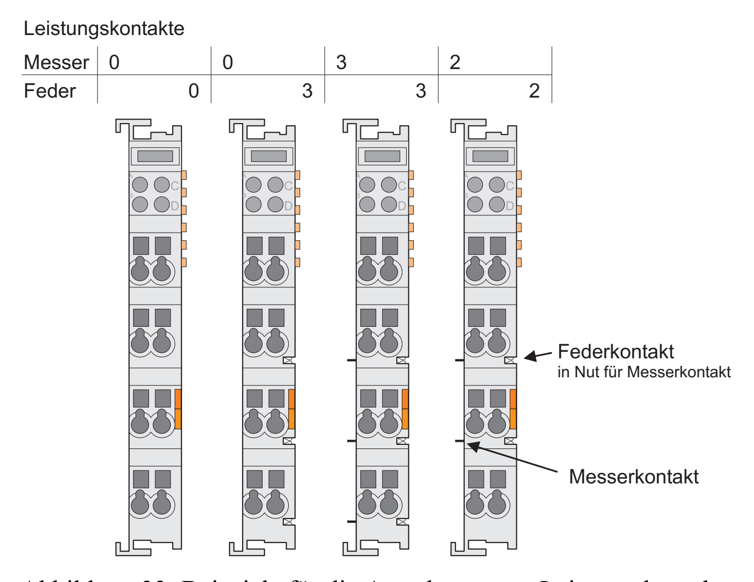
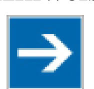


Abbildung 33: Beispiele für die Anordnung von Leistungskontakten

Hinweis



Feldbusknoten mit smartDESIGNER konfigurieren und überprüfen
Sie können mit der WAGO-ProServe®-Software **smartDESIGNER** den Aufbau eines Feldbusknotens konfigurieren. Über die integrierte Plausibilitätsprüfung können Sie die Konfiguration überprüfen.

6.3 Leiter an CAGE CLAMP® anschließen

CAGE CLAMP®-Anschlüsse von WAGO sind für ein-, mehr- oder feindrähtige Leiter ausgelegt.

Hinweis



Nur einen Leiter pro CAGE CLAMP® anschließen!
Sie dürfen an jedem CAGE CLAMP®-Anschluss nur einen Leiter anschließen. Mehrere einzelne Leiter an einem Anschluss sind nicht zulässig.

Müssen mehrere Leiter auf einen Anschluss gelegt werden, verbinden Sie diese in einer vorgelagerten Verdrahtung, z. B. mit WAGO-Durchgangsklemmen.

Ausnahme:
Sollte es unvermeidbar sein, zwei mehr- oder feindrähtige Leiter an einem CAGE CLAMP®-Anschluss anzuschließen, müssen Sie eine gemeinsame Aderendhülse verwenden. Folgende Aderendhülsen sind einsetzbar:

Länge	8 mm
Nennquerschnitt $_{max.}$	1 mm² für zwei mehr- oder feindrähtige Leiter mit je 0,5 mm²
WAGO-Produkt	216-103 oder Produkte mit gleichen Eigenschaften.

1. Zum Öffnen der CAGE CLAMP® führen Sie das Betätigungswerkzeug in die Öffnung oberhalb des Anschlusses ein.

2. Führen Sie den Leiter in die entsprechende Anschlussöffnung ein.

3. Zum Schließen der CAGE CLAMP® entfernen Sie das Betätigungswerkzeug wieder. Der Leiter ist festgeklemmt.

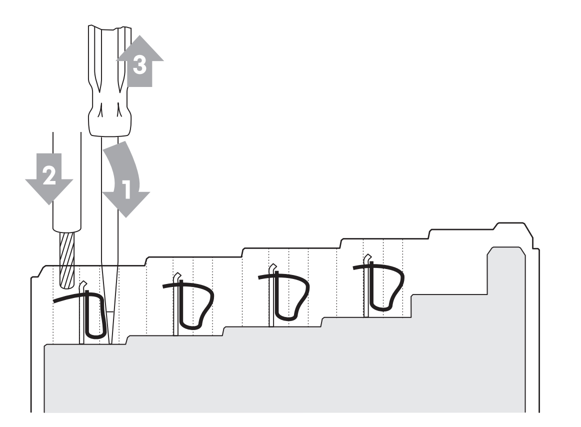


Abbildung 34: Leiter an CAGE CLAMP® anschließen



7 Funktionsbeschreibung

7.1 Betriebssystem

7.1.1 Anlauf des Feldbuscontrollers

Hinweis



Betriebsartenschalter darf sich nicht in der unteren Stellung befinden!
Damit ein Anlauf erfolgen kann, darf der Betriebsartenschalter beim Anlauf nicht in die untere Stellung geschaltet sein!

Nach Einschalten der Versorgungsspannung oder nach Hardware-Reset startet der Feldbuscontroller.

Das intern vorhandene PFC-Programm wird ins RAM übertragen.

In der Initialisierungsphase ermittelt der Feldbuscontroller die Busklemmen und die vorliegende Konfiguration und setzt die Variablen auf 0 bzw. auf FALSE oder auf einen von dem PFC-Programm vorgegebenen Initialwert.
Die Merker behalten ihren Zustand bei.
Während dieser Phase blinkt die I/O-LED rot.

Nach fehlerfreiem Anlauf leuchtet die I/O-LED grün.

7.1.2 PFC-Zyklus

Nach fehlerfreiem Anlauf startet der PFC-Zyklus bei oberer Stellung des Betriebsartenschalters oder durch einen Start-Befehl aus WAGO-I/O-*PRO* CAA.
Die Ein- und Ausgangsdaten des Feldbusses und der Busklemmen sowie die Werte von Zeitgebern werden gelesen. Anschließend wird das im RAM vorhandene PFC-Programm bearbeitet und danach die Ausgangsdaten des Feldbusses und der Busklemmen ins Prozessabbild geschrieben. Am Ende des PFC-Zyklus werden Betriebssystemfunktionen u. a. für Diagnose und Kommunikation ausgeführt und die Werte von Zeitgebern aktualisiert.
Der Zyklus beginnt erneut mit dem Einlesen der Ein- und Ausgangsdaten und der Werte von Zeitgebern.

Der Wechsel der Betriebsart („STOP“/“RUN“) erfolgt am Ende eines PFC-Zyklus.

Die Zykluszeit ist die Zeit vom Beginn des PFC-Programms bis zum nächsten Beginn. Wenn innerhalb eines PFC-Programms eine Schleife programmiert wird, verlängert sich entsprechend die PFC-Laufzeit und somit der PFC-Zyklus.

Während der Bearbeitung des PFC-Programms werden die Eingänge, Ausgänge und Werte von Zeitgebern nicht aktualisiert. Diese Aktualisierung findet erst definiert am Ende des PFC-Programms statt. Hieraus ergibt sich, dass es nicht möglich ist, innerhalb einer Schleife auf ein Ereignis aus dem Prozess oder den Ablauf einer Zeit zu warten.

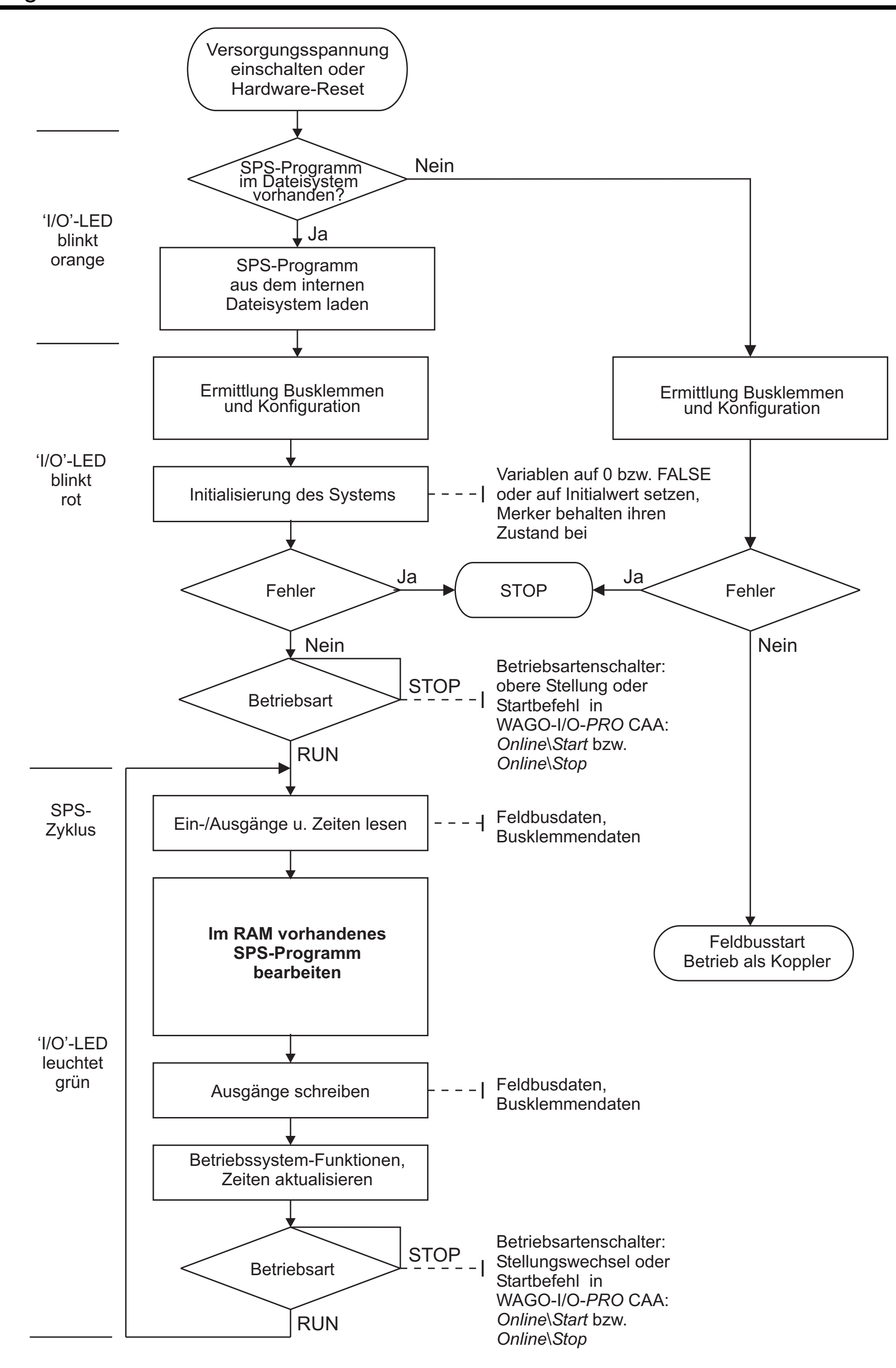


Abbildung 35: Anlauf des Feldbuscontrollers

7.2 Prozessdatenaufbau

7.2.1 Prinzipieller Aufbau

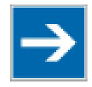
Nach dem Einschalten erkennt der Controller alle im Knoten gesteckten Busklemmen, die Daten liefern bzw. erwarten (Datenbreite/Bitbreite > 0).
In einem Knoten können analoge und digitale Klemmen gemischt angeordnet sein.

Hinweis



Mit Klemmenbusverlängerung bis zu 250 Busklemmen anschließbar!
Mit dem Einsatz der WAGO-Klemmenbusverlängerungs-Kopplerklemme 750-628 und -Endklemme 750-627 ist es möglich, an dem Feldbuscontroller bis zu 250 Busklemmen zu betreiben.

Hinweis



Erweiterung auf 250 Busklemmen ab Software-Version 9!
Die Erweiterung auf 250 Busklemmen ist für den Feldbuscontroller ab Software-Version 9 freigegeben.

Information



Weitere Information
Die Anzahl der Ein- und Ausgangsbits bzw. -bytes der einzeln angeschalteten Busklemmen entnehmen Sie den entsprechenden Beschreibungen der Busklemmen.

Aus der Datenbreite und dem Typ der Busklemme sowie der Position der Busklemmen im Knoten erstellt der Controller ein internes lokales Prozessabbild. Es ist in einen Eingangs- und Ausgangsdatenbereich unterteilt.

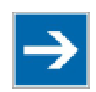
Die Daten der digitalen Busklemmen sind bitorientiert, d. h. der Datenaustausch erfolgt bitweise. Die analogen Busklemmen stehen stellvertretend für alle byteorientierten Busklemmen, bei denen der Datenaustausch also byteweise erfolgt.

Zu diesen Busklemmen gehören z. B. die Zählerklemmen, Busklemmen für Winkel- und Wegmessung sowie die Kommunikationsklemmen.

Für das lokale Ein- und Ausgangsprozessabbild werden die Daten der Busklemmen in der Reihenfolge ihrer Position am Controller in dem jeweiligen Prozessabbild abgelegt.

Dabei werden zuerst die byteorientierten (analogen) und im Anschluss daran die bitorientierten (digitalen) Busklemmen im Prozessabbild abgelegt. Die Bits der digitalen Klemmen werden zu Bytes zusammengefügt. Ist die Anzahl der digitalen Eingänge/Ausgänge größer als 8 Bit, beginnt der Controller automatisch ein weiteres Byte.

Hinweis



Hardware-Änderung kann Änderung des Prozessabbildes bewirken!
Wenn die Hardware-Konfiguration durch Hinzufügen, Austausch oder Entfernen von Busklemmen mit einer Datenbreite > 0 Bit geändert wird, ergibt sich daraus ein neuer Aufbau des Prozessabbildes. Damit ändern sich auch die Adressen der Prozessdaten. Bei einer Erweiterung sind die Prozessdaten aller vorherigen Klemmen zu berücksichtigen.

Für das Prozessabbild der physikalischen Ein- und Ausgangsdaten steht in dem Controller zunächst jeweils ein Speicherbereich von 256 Worten (Wort 0...255) zur Verfügung.

Für die Abbildung der MODBUS/PFC-Variablen ist der Speicherbereich von jeweils Wort 256...511 reserviert, so dass die MODBUS/PFC-Variablen hinter dem Prozessabbild der Busklemmendaten abgebildet werden.

Ist die Anzahl der Klemmendaten größer als 256 Worte, werden alle darüber hinausreichenden physikalischen Ein- und Ausgangsdaten in einem Speicherbereich an das Ende des bisherigen Prozessabbildes und somit hinten an die MODBUS/PFC-Variablen angehängt (Wort 512...1275).

Im Anschluss an die restlichen physikalischen Busklemmendaten werden die Ethernet IP PFC-Variablen abgebildet. Dieser Speicherbereich umfasst Wort 1276...1531.

Für zukünftige Protokoll-Erweiterungen ist der anschließende Bereich ab Wort 1532 für die weiteren PFC-Variablen reserviert.

Bei allen WAGO-Feldbuscontrollern ist der Zugriff der SPS auf die Prozessdaten unabhängig von dem Feldbussystem. Dieser Zugriff erfolgt stets über ein anwendungsbezogenes IEC-61131-3-Programm.
Der Zugriff von der Feldbusseite aus ist dagegen feldbusspezifisch.

Für den Feldbuscontroller kann ein MODBUS/TCP-Master über implementierte MODBUS-Funktionen auf die Daten zugreifen, wobei dezimale, bzw. hexadezimale MODBUS-Adressen verwendet werden.
Wahlweise kann der Datenzugriff auch über Ethernet/IP mittels eines Objektmodells erfolgen.



Information



Weitere Information
Eine detaillierte Beschreibung zu diesen feldbusspezifischen Datenzugriffen finden Sie in dem Kapitel „MODBUS-Funktionen“ bzw. in dem Kapitel „Ethernet/IP (Ethernet/Industrial Protocol)“.

Information



Weitere Information
Das feldbusspezifische Prozessabbild ist in dem Kapitel „Aufbau der Prozessdaten“ für jede WAGO-Busklemme im Einzelnen dargestellt.



7.2.2 Beispiel für ein Eingangsprozessabbild

Im folgenden Bild wird ein Beispiel für ein Prozessabbild mit Eingangsklemmendaten dargestellt.
Die Konfiguration besteht aus 16 digitalen und 8 analogen Eingängen.
Das Eingangsprozessabbild hat damit eine Datenlänge von 8 Worten für die analogen Klemmen und 1 Wort für die digitalen, also insgesamt 9 Worte.

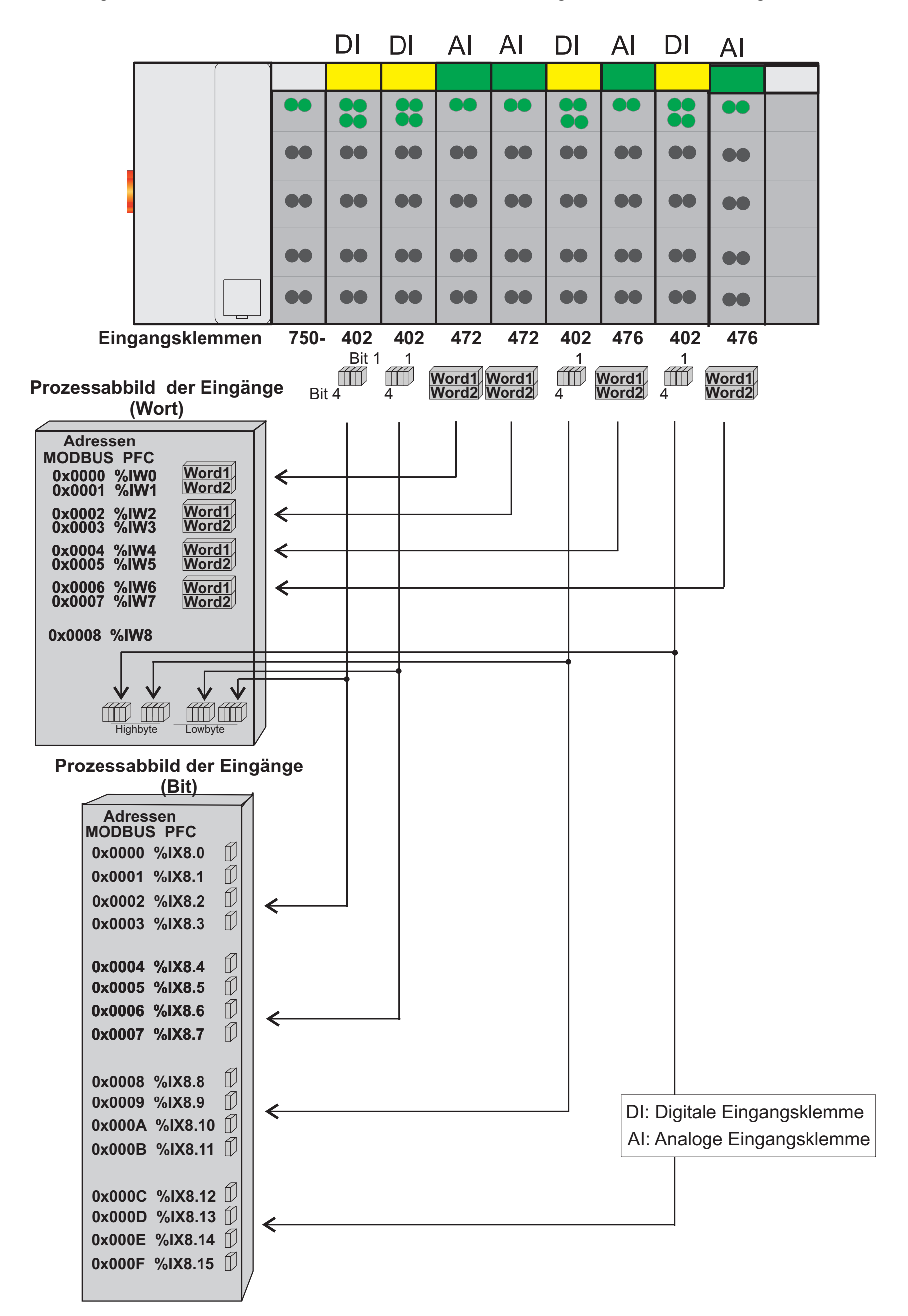


Abbildung 36: Beispiel Eingangsprozessabbild

7.2.3 Beispiel für ein Ausgangsprozessabbild

Als Beispiel für das Prozessabbild mit Ausgangsklemmendaten besteht die folgende Konfiguration aus 2 digitalen und 4 analogen Ausgängen.
Das Ausgangsdaten Prozessabbild besteht aus 4 Worten für die analogen und einem Wort für die digitalen Ausgänge, also insgesamt aus 5 Worten.

Zusätzlich können die Ausgangsdaten mit einem auf die MODBUS-Adresse aufaddierten Offset von 200_{hex} (0x0200) zurückgelesen werden.

Hinweis



Daten > 256 Worte sind mittels aufaddiertem Offset rücklesbar!
Alle Ausgangsdaten, die über 256 Worte hinausreichen und sich deshalb im Speicherbereich 6000_{hex} (0x6000) bis $66F9_{hex}$ (0x66F9) befinden, können mit einem auf die MODBUS-Adresse aufaddierten Offset von 1000_{hex} (0x1000) zurückgelesen werden.



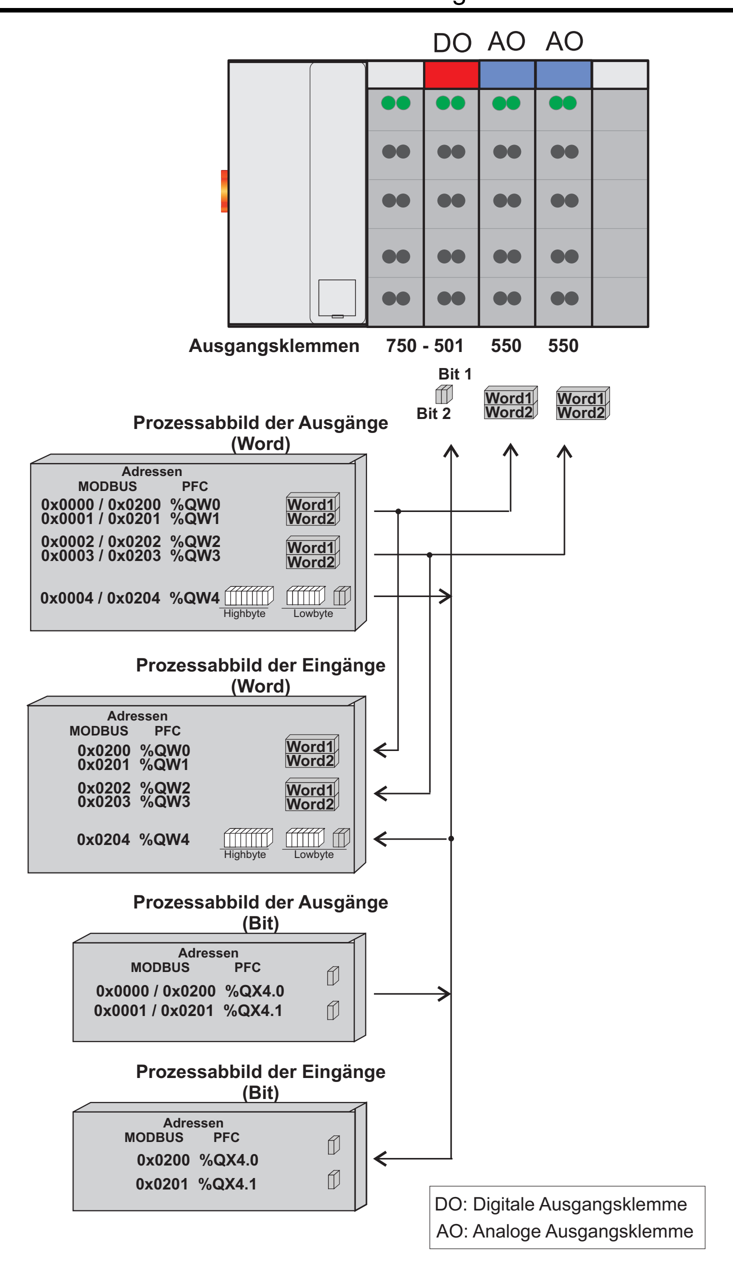


Abbildung 37: Beispiel Ausgangsprozessabbild

7.2.4 Prozessdaten MODBUS/TCP und EtherNet/IP

Der Aufbau der Prozessdaten ist auf der Feldebene bei einigen Busklemmen bzw. deren Varianten feldbusspezifisch.

Bei dem Feldbuscontroller mit MODBUS und EtherNet/IP wird das Prozessabbild wortweise aufgebaut (mit word-alignment). Die interne Darstellung der Daten, die größer als ein Byte sind, erfolgt nach dem Intel-Format.

Information



Weitere Information zu dem feldbusspezifischen Prozessdatenaufbau
Der entsprechende feldbusspezifische Aufbau der Prozesswerte aller Busklemmen des WAGO-I/O-SYSTEM 750 und 753 finden Sie in dem Kapitel “Aufbau der Prozessdaten für MODBUS/TCP” bzw. “Aufbau der Prozessdaten für EtherNet/IP”.



7.3 Datenaustausch

Der Austausch der Prozessdaten findet bei dem Feldbuscontroller entweder über das MODBUS/TCP-Protokoll bzw. MODBUS/UDP-Protokoll oder über Ethernet/IP statt.

MODBUS/TCP arbeitet nach dem Master-/Slave-Prinzip. Der Master ist eine übergeordnete Steuerung, z. B. ein PC oder eine Speicherprogrammierbare Steuerung.

Die Feldbuscontroller des WAGO-I/O-SYSTEM 750 sind in der Regel Slavegeräte. Durch die Programmierung mit IEC 61131-3 können aber auch Controller zusätzlich die Master-Funktion übernehmen.

Der Master fordert die Kommunikation an. Diese Anforderung kann durch die Adressierung an einen bestimmten Knoten gerichtet sein. Die Knoten empfangen die Anforderung und senden, abhängig von der Art der Anforderung, eine Antwort an den Master.

Ein Feldbuscontroller kann eine bestimmte Anzahl gleichzeitiger Verbindungen (Socket-Verbindungen) zu anderen Netzwerkteilnehmern herstellen:

- 3 Verbindung für HTTP (HTML-Seiten von dem Controller lesen),
- 15 Verbindungen über MODBUS/TCP (Ein- und Ausgangsdaten vom Controller lesen oder schreiben),
- 128 Ethernet IP Verbindungen,
- 5 Verbindungen über den PFC (verfügbar in der SPS-Funktionalität für IEC 61131-3 Applikationsprogramme) und
- 2 Verbindungen für WAGO-I/O-*PRO* CAA
  (Diese Verbindungen sind reserviert für das Debuggen des Applikationsprogramms über ETHERNET. WAGO-I/O-*PRO* CAA benötigt für das Debuggen 2 Verbindungen zur selben Zeit. Es kann jedoch nur ein Programmiertool Zugriff auf den Controller haben.)
- 10 Verbindungen für FTP
- 2 Verbindungen für SNMP

Die maximale Anzahl der gleichzeitigen Verbindungen kann nicht überschritten werden. Sollen weitere Verbindungen aufgebaut werden, müssen bestehende Verbindungen vorher beendet werden.

Für den Austausch von Daten besitzt der Feldbuscontroller im Wesentlichen drei Schnittstellen:

- die Schnittstelle zum Feldbus (Feldbusmaster),
- die SPS-Funktionalität des Feldbuscontrollers (CPU)
- die Schnittstelle zu den Busklemmen

Es findet ein Datenaustausch zwischen Feldbusmaster und den Busklemmen, zwischen SPS-Funktionalität des Feldbuscontrollers (CPU) und den Busklemmen und zwischen Feldbusmaster und SPS-Funktionalität des PFCs (CPU) statt.

Wird der Feldbus MODBUS genutzt, greift der MODBUS-Master über die in dem Controller implementierten MODBUS-Funktionen auf Daten zu, Ethernet/IP hingegen verwendet für den Datenzugriff ein Objektmodell.

Der Zugriff des Feldbuscontrollers auf die Daten erfolgt mit Hilfe eines IEC-61131-3-Applikationsprogramms. Die Adressierung der Daten ist dabei jeweils sehr unterschiedlich.

7.3.1 Speicherbereiche

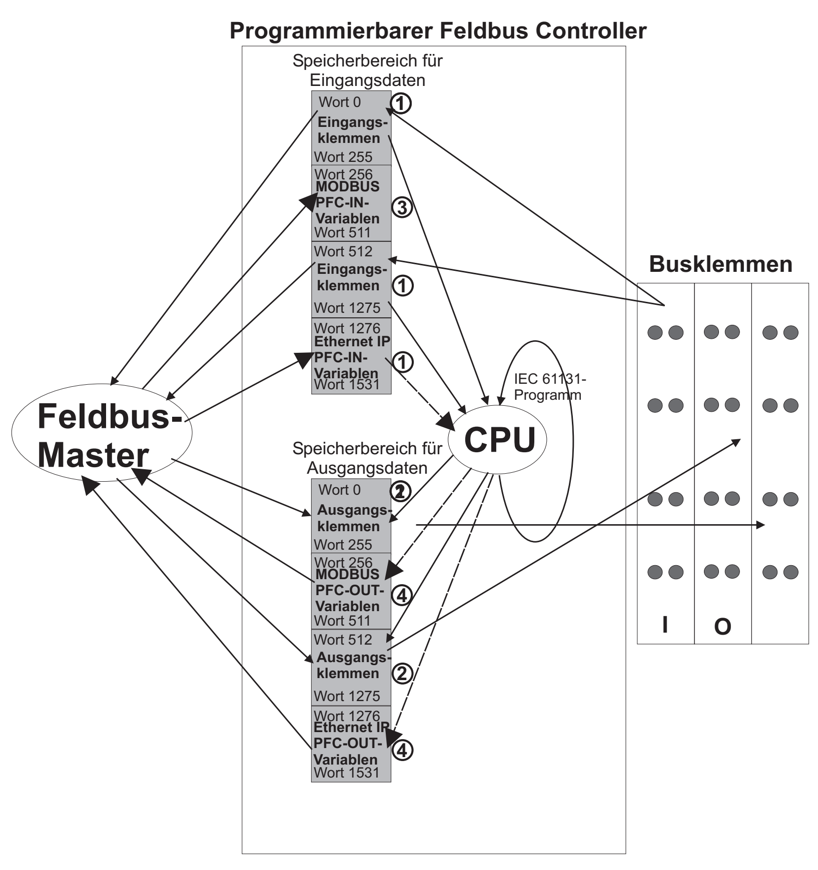


Abbildung 38: Speicherbereiche und Datenaustausch

Das Prozessabbild des Controllers beinhaltet die physikalischen Daten der Busklemmen. Diese belegen im Speicherbereich Wort 0...255 und Wort 512...1275.



① Von der CPU und von der Feldbusseite können die Eingangsklemmendaten gelesen werden.

② Ebenso kann von Seite der CPU und Feldbusseite auf die Ausgangsklemmen geschrieben werden.

In dem jeweils dazwischen liegenden Speicherbereich Wort 256...511 sind die MODBUS-PFC-Variablen abgelegt.

③ Von der Feldbusseite werden die MODBUS-PFC-Eingangsvariablen in den Eingangsspeicherbereich geschrieben und von der CPU zur Verarbeitung eingelesen.

④ Die von der CPU über das IEC-61131-3-Programm verarbeiteten Variablen werden in den Ausgangsspeicherbereich gelegt und können von dem Master ausgelesen werden.

Im Anschluss an die physikalischen Busklemmendaten befindet sich der Speicherbereich Wort 1276...1531 für die Ethernet/IP-PFC-Variablen.

Für zukünftige Protokoll-Erweiterungen und weitere PFC-Variablen ist der anschließende Speicherbereich ab Wort 1532 vorgesehen.

Zusätzlich sind alle Ausgangsdaten auf einen Speicherbereich mit dem Adressen-Offset 0x0200 bzw. 0x1000 gespiegelt. Dadurch ist es möglich, durch Hinzuaddieren von 0x0200 bzw. 0x1000 zu der MODBUS-Adresse Ausgangswerte zurückzulesen.

In dem Controller sind darüber hinaus weitere Speicherbereiche vorhanden, auf die teilweise von der Feldbusseite aus jedoch nicht zugegriffen werden kann:

- **Datenspeicher (256 kByte)**
  Der Datenspeicher ist ein flüchtiger RAM-Speicher und dient zum Anlegen von Variablen, die nicht zur Kommunikation mit den Schnittstellen sondern für interne Verarbeitungen, wie z. B. die Berechnung von Ergebnissen benötigt werden.

- **Programmspeicher (512 kByte)**
  In dem Programmspeicher wird das IEC-61131-3-Programm abgelegt. Der Code-Speicher ist ein Flash-ROM. Nach dem Einschalten der Versorgungsspannung wird das Programm von dem Flash- in den RAM-Speicher übertragen. Nach fehlerfreiem Hochlauf startet der PFC-Zyklus bei oberer Stellung des Betriebsartenschalters oder durch einen Startbefehl aus WAGO-I/O-*PRO* CAA.

- **NOVRAM Remanentspeicher (24 kByte)**
  Der Remanentspeicher ist ein nicht flüchtiger Speicher, d. h. nach einem Spannungsausfall bleiben alle Werte der Merker und Variablen beibehalten, die explizit mit „var retain“ definiert werden. Die Speicherverwaltung erfolgt automatisch. Der 24 kByte große Speicherbereich teilt sich standardmäßig in einen 8 kByte großen adressierbaren Bereich für die

Merker (%MW0 ... %MW4095) und einen 16 kByte großen Retain-Bereich für Variablen ohne Speicherbereichs-Adressierung, die mit „var retain" definiert werden.

Hinweis



Merker nur unter „var retain" remanent!
Beachten Sie, dass die Merker nur remanent sind, wenn Sie diese unter „var retain" deklariert haben.

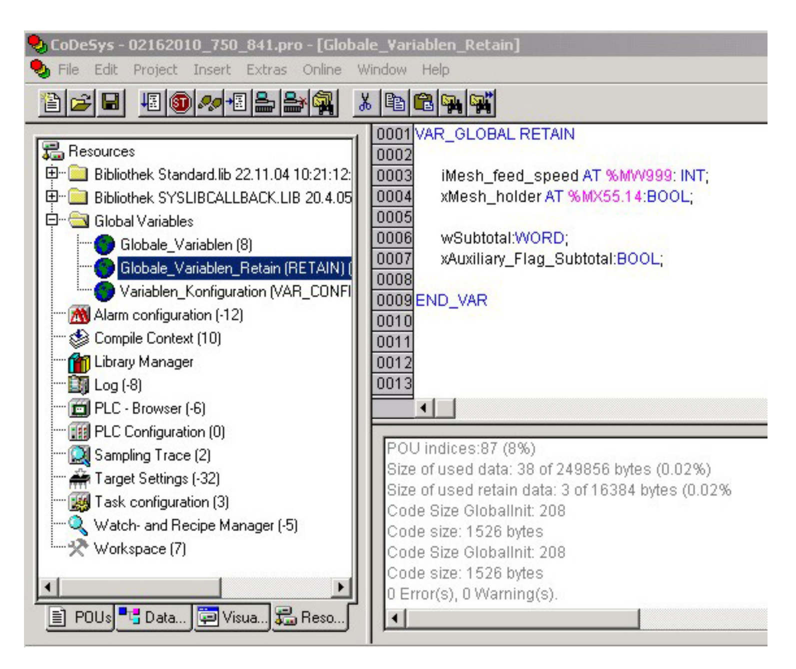


Abbildung 39: Beispieldeklarierung für remanente Merker unter „var retain"

Die Aufteilung des NOVRAM Remanentspeichers ist variabel (siehe nachfolgenden Hinweis).

Hinweis



NOVRAM-Speicheraufteilung in WAGO-I/O-*PRO* CAA änderbar!
Die Aufteilung des NOVRAM ist in der Programmiersoftware WAGO-I/O-*PRO* CAA/Register „Ressourcen"/Dialogfenster „Zielsystem Einstellungen" bei Bedarf veränderbar.
Die Startadresse für den Merker-Bereich ist dabei fest adressiert mit 16#30000000. Die Bereichsgrößen und die Startadresse des Retain-Speichers sind variabel.
Um eine Überlappung der Bereiche auszuschließen, wird jedoch empfohlen, die Standardeinstellung beizubehalten.
Hierbei ist die Größe des Merker-Bereichs mit 16#2000 vorgegeben und daran im Anschluss der Retain-Speicher mit der Startadresse 16#30002000 und der Größe 16#4000.

7.3.2 Adressierung

Ein- und Ausgänge der Klemmen an einem Controller werden intern adressiert, sobald sie in Betrieb genommen werden. Die Reihenfolge, in welcher die



gesteckten Klemmen adressiert werden, hängt von der Art der Klemme (Eingangsklemme, Ausgangsklemme etc.) ab. Aus diesen Adressen baut sich das Prozessabbild zusammen.

Hinweis



Verschiedene Möglichkeiten zur Adressierung der Busklemmen nutzen!
In diesem Kapitel wird die Adressierung und interne Funktionsweise eines Feldbuscontrollers mit gesteckten Klemmen näher erläutert. Ein Verständnis der Zusammenhänge ist wichtig, wenn Sie Adressen konventionell über Auszählen zuweisen möchten.
Neben dieser Möglichkeit der Adressierung steht Ihnen der **WAGO-I/O-Konfigurator** zur Verfügung. Dieser unterstützt Sie bei der Adressierung und Protokollzuweisung der gesteckten Klemmen. Sie wählen im I/O-Konfigurator die gesteckten Klemmen aus.
Die korrekte Adressierung übernimmt die Software für Sie.

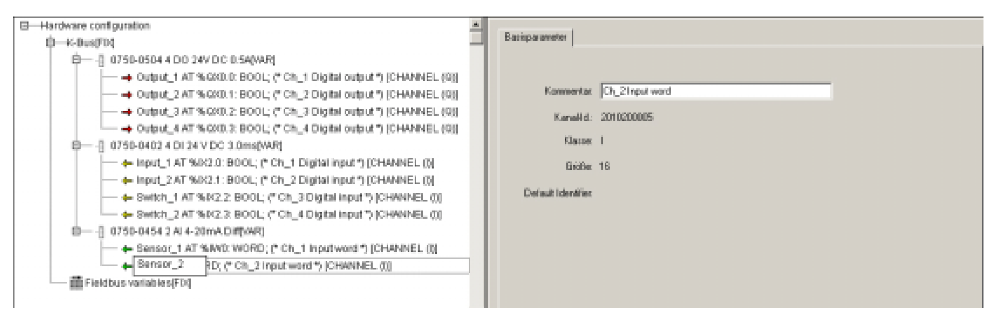


Abbildung 40: WAGO-I/O-Konfigurator

Der I/O-Konfigurator wird aus der WAGO-I/O-*PRO* CAA heraus gestartet. Eine nähere Beschreibung lesen Sie in Kapitel „Feldbuscontroller mit dem I/O-Konfigurator konfigurieren".

7.3.2.1 Adressierung der Busklemmen

Bei der Adressierung werden zunächst die komplexen Klemmen (Klemmen, die ein oder mehrere Byte belegen) entsprechend ihrer physikalischen Reihenfolge hinter dem Feldbuskoppler/-controller berücksichtigt. Diese belegen somit die Adressen ab Wort 0.

Im Anschluss daran folgen, immer in Bytes zusammengefasst, die Daten der übrigen Busklemmen (Klemmen, die weniger als ein Byte belegen). Dabei wird entsprechend der physikalischen Reihenfolge Byte für Byte mit diesen Daten aufgefüllt. Sobald ein ganzes Byte durch die bitorientierten Klemmen belegt ist, wird automatisch das nächste Byte begonnen.

Hinweis



Hardware-Änderung kann Änderung des Prozessabbildes bewirken!
Wenn die Hardware-Konfiguration geändert bzw. erweitert wird, kann sich daraus ein neuer Aufbau des Prozessabbildes ergeben. Damit ändern sich dann auch die Adressen der Prozessdaten. Bei einer Erweiterung sind die Prozessdaten aller vorherigen Klemmen zu berücksichtigen.

Hinweis



Prozessdatenanzahl beachten!
Entnehmen Sie die Anzahl der Ein- und Ausgangsbits bzw. -bytes für die einzelnen angeschalteten Busklemmen den entsprechenden Beschreibungen der Busklemmen.

Tabelle 27: Datenbreite der Busklemmen

Datenbreite ≥ 1 Wort/Kanal	Datenbreite = 1 Bit/Kanal
Analoge Eingangsklemmen	Digitale Eingangsklemmen
Analoge Ausgangsklemmen	Digitale Ausgangsklemmen
Eingangsklemmen für Thermoelemente	Digitale Ausgangsklemmen mit Diagnose (2 Bit/Kanal)
Eingangsklemmen für Widerstandssensoren	Einspeiseklemmen mit Sicherungshalter/Diagnose
Pulsweiten Ausgangsklemmen	Solid State Lastrelais
Schnittstellenklemmen	Relais-Ausgangsklemmen
Vor-/Rückwärtszähler	
Busklemmen für Winkel- und Wegmessung	

7.3.2.2 Adressbereiche

Aufteilung des Adressbereiches für die wortweise Adressierung nach IEC-61131-3:

Tabelle 28: Aufteilung des Adressbereiches

Wort	Daten
0-255	Physikalische Busklemmen
256-511	MODBUS-PFC-Variablen
512-1275	Weitere physikalische Busklemmen
1276-1531	Ethernet/IP PFC-Variablen
1532-...	Für PFC-Variablen zukünftiger Protokolle vorgesehen

Wort 0...255: Erster Adressbereich für die Ein-/Ausgangsdaten der Busklemmen:

Tabelle 29: Adressbereich Wort 0...255

Daten-breite	Adresse								
Bit	0.0... 0.7	0.8... 0.15	1.0... 1.7	1.8... 1.15	...	254.0... 254.7	254.8... 254.15	255.0... 255.7	255.8... 255.15
Byte	0	1	2	3	...	508	509	510	511
Word	**0**		**1**		**...**	**254**		**255**	
DWord	0				...	127			



Wort 256...511: Adressbereich für die MODBUS-PFC-Variablen:

Tabelle 30: Adressbereich Wort 256...511

Daten-breite	**Adresse**								
Bit	256..0 ... 256.7	256..8 ... 256.15	257.0 ... 257.7	257.8 ... 257.15	...	510.0 ... 510.7	510.8 ... 510.15	511.0 ... 511.7	511.8 ... 511.15
Byte	512	513	514	515	...	1020	1021	1022	1023
Word	**256**		**257**		**...**	**510**		**511**	
DWord	128				...	255			

Wort 512...1275: Zweiter Adressbereich für die Ein-/Ausgangsdaten der Busklemmen:

Tabelle 31: Adressbereich Wort 512...1275

Daten-breite	**Adresse**								
Bit	512.0 ... 512.7	512.8 ... 512.15	513.0 ... 513.7	513.8 ... 513.15	...	1274.0 ... 1274.7	1274.8 ... 1274.15	1275.0 ... 1275.7	1275.8 ... 1275.15
Byte	1024	1025	1026	1027	...	2548	2549	2550	2551
Word	**512**		**513**		**...**	**1274**		**1275**	
DWord	256				...	637			

Wort 1276-1531: Adressbereich für die Ethernet/IP-Feldbusdaten:

Tabelle 32: Adressbereich Wort 1276...1531

Datenb reite	**Adresse**								
Bit	1276.0. 1276.7	1276.8. 1276.15	1277.0. 1277.7	1277.8. 1277.15	.....	1530.0...153 0.7	1530.8...153 0.15	1531.0...153 1.7	1531.8...153 1.15
Byte	2552	2553	2554	2555	.....	3060	3061	3062	3063
Word	**1276**		**1277**		.....	**1530**		**1531**	
DWord	638				.....	765			

Adressbereich für Merker:

Tabelle 33: Adressbereich für Merker

Daten-breite	**Adresse**								
Bit	0.0 ... 0.7	0.8 ... 0.15	1.0 ... 1.7	1.8 ... 1.15	...	12287.0 ... 12287.7	12287.8 ... 12287.15	12288.0 ... 12288.7	12288.8 ... 12288.15
Byte	0	1	2	3	...	24572	24573	24574	24575
Word	**0**		**1**		**...**	**12287**		**12288**	
DWord	0				...	6144			

IEC-61131-3-Adressräume in der Übersicht:

Tabelle 34: IEC-61131-3-Adressräume

Adressraum	**MODBUS-Zugriff**	**SPS-Zugriff**	**Beschreibung**
phys. Eingänge	read	read	Physikalische Eingänge (%IW0...%IW255 und %IW512...%IW1275)
phys. Ausgänge	read/write	read/write	Physikalische Ausgänge (%QW0...%QW255 und %QW512...%QW1275)
MODBUS/TCP PFC-IN-Variablen	read/write	read	Flüchtige SPS-Eingangsvariablen (%IW256...%IW511)
MODBUS/TCP PFC-OUT-Variablen	read	read/write	Flüchtige SPS-Ausgangsvariablen (%QW256...%QW511)
Ethernet/IP-PFC-IN-Variablen	-	read	Flüchtige SPS-Eingangsvariablen (%IW1276 ... %IW1531)
Ethernet/IP-PFC-OUT-Variablen	-	read/write	Flüchtige SPS-Ausgangsvariablen (%QW1276 ... %QW1531)
Konfigurationsregister	read/write	-	siehe Kapitel „MODBUS-Funktionen →MODBUS-Register → Konfigurationsregister“
Firmware-Register	read	-	siehe Kapitel „MODBUS-Funktionen → MODBUS-Register → Firmware-Informationsregister“
Retain-Variablen	read/write	read/write	Remanent-Speicher (%MW0...%MW12288)

7.3.2.3 Absolute Adressierung

Die direkte Darstellung einzelner Speicherzellen (absolute Adressen) nach IEC-61131-3 erfolgt mittels spezieller Zeichenketten:



Tabelle 35: Absolute Adressen

Position	Zeichen	Benennung	Kommentar
1	%	Leitet absolute Adresse ein	
2	I Q M	Eingang Ausgang Merker	
3	X* B W D	Einzelbit Byte (8 Bits) Word (16 Bits) Doubleword (32 Bits)	Datenbreite
4		Adresse	

z. B. wortweise: %QW27 (28. Wort), bitweise: %IX1.9 (10.Bit im 2. Wort)
* Das Kennzeichen ‘X’ für Bits kann entfallen

Hinweis



Zeichenketten ohne Leer- und Sonderzeichen eingeben!
Die Zeichenketten der absoluten Adressen sind zusammenhängend, d. h. ohne Leerzeichen oder Sonderzeichen einzugeben!

Beispieladressierungen:

Tabelle 36: Beispieladressierung

	Eingänge			
Bit	%IX14.0 ... 15		%IX15.0 ... 15	
Byte	%IB28	%IB29	%IB30	%IB31
Wort	**%IW14**		**%IW15**	
Doppelwort	%ID7			

	Ausgänge			
Bit	%QX5.0 ... 15		%QX6.0 ... 15	
Byte	%QB10	%QB11	%QB12	%QB13
Wort	**%QW5**		**%QW6**	
Doppelwort	%QD2 (oberer Teil)		%QD3 (unterer Teil)	

	Merker			
Bit	%MX11.0 ... 15		%MX12.0 ... 15	
Byte	%MB22	%MB23	%MB24	%MB25
Wort	**%MW11**		**%MW12**	
Doppelwort	%MD5 (oberer Teil)		%MD6 (unterer Teil)	

Adressen berechnen (in Abhängigkeit von der Wortadresse):

Bit-Adresse: Wortadresse .0 bis .15
Byte-Adresse: 1. Byte: 2 x Wortadresse

2. Byte: 2 x Wortadresse + 1

DWord-Adresse: Wortadresse (gerade Zahl) / 2
bzw. Wortadresse (ungerade Zahl) / 2, abgerundet

7.3.3 Datenaustausch MODBUS/TCP-Master und Busklemmen

Der Datenaustausch zwischen MODBUS/TCP-Master und den Busklemmen erfolgt über die in dem Feldbuskoppler/-controller implementierten MODBUS-Funktionen durch bit- oder wortweises Lesen und Schreiben.

Im Feldbuskoppler/-controller gibt es 4 verschiedene Typen von Prozessdaten:

- Eingangsworte
- Ausgangsworte
- Eingangsbits
- Ausgangsbits

Der wortweise Zugriff auf die digitalen Ein- und Ausgangsklemmen erfolgt entsprechend der folgenden Tabelle:

Tabelle 37: Zuordnung digitale Ein-/Ausgänge zum Prozessdatenwort gemäß Intel-Format

Digitale Eingänge/ Ausgänge	16.	15.	14.	13.	12.	11.	10.	9.	8.	7.	6.	5.	4.	3.	2.	1.
Prozessdaten-wort	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte	High-Byte D1								Low-Byte D0							

Durch Hinzuaddieren eines Offsets von 200 hex (0x0200) zu der MODBUS-Adresse können die Ausgänge zurückgelesen werden.

Hinweis



Daten > 256 Worte sind mittels aufaddiertem Offset rücklesbar!
Alle Ausgangsdaten, die über 256 Worte hinausreichen und deshalb in dem Speicherbereich 0x6000 bis 0x62FC liegen, können mit einem auf die MODBUS-Adresse aufaddierten Offset von 1000_{hex} (0x1000) zurückgelesen werden.



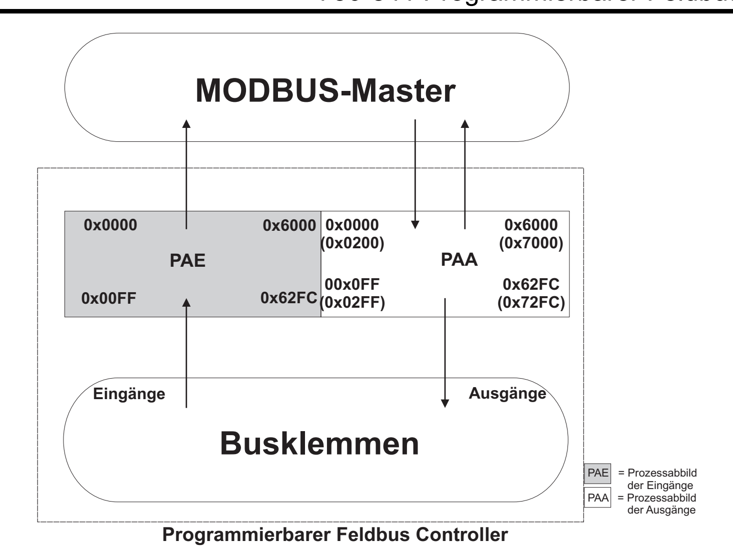


Abbildung 41: Datenaustausch zwischen MODBUS-Master und Busklemmen

Ab Adresse 0x1000 liegen die Registerfunktionen. Diese sind analog mit den implementierten MODBUS-Funktionscodes (read/write) ansprechbar. Anstatt der Adresse eines Klemmenkanals wird dazu die jeweilige Registeradresse angegeben.

Information



Weitere Information
Eine detaillierte Beschreibung der MODBUS-Adressierung ist in dem Kapitel „MODBUS-Register-Mapping“ zu finden.

7.3.3.1 Datenaustausch EtherNet/IP-Master und Busklemmen

Der Datenaustausch zwischen EtherNet/IP-Master und den Busklemmen ist objektorientiert. Jeder Knoten im Netz wird als Sammlung von Objekten dargestellt.

Das Assembly-Object legt den Aufbau der Objekte für die Datenübertragung fest. Mit dem Assembly-Object können Daten (z. B. I/O-Daten) zu Blöcken zusammengefasst (gemappt) und über eine einzige Nachrichtenverbindung versendet werden. Durch dieses Mapping sind weniger Zugriffe auf das Netzwerk nötig.

Es wird zwischen In- und Output-Assemblies unterschieden.

Eine Input-Assembly liest Daten von der Applikation über das Netz ein bzw. produziert Daten auf dem Netzwerk.

Eine Output-Assembly schreibt Daten an die Applikation bzw. konsumiert Daten vom Netzwerk.

In dem Feldbuskoppler/-controller sind bereits verschiedene Assembly-Instanzen fest vorprogrammiert (statisches Assembly).

Nach Einschalten der Versorgungsspannung werden von dem Assembly-Object Daten aus dem Prozessabbild zusammengefasst. Sobald eine Verbindung aufgebaut ist, kann der Master die Daten mit „Klasse", „Instanz" und „Attribut" adressieren und darauf zugreifen, bzw. mittels I/O-Verbindungen lesen und/oder schreiben.

Das Mapping der Daten ist abhängig von der gewählten Assembly-Instanz des statischen Assembly.

Information



Weitere Information
Die Assembly-Instanzen für das statische Assembly sind in dem Kapitel „EtherNet/IP" beschrieben.

7.3.4 Datenaustausch SPS-Funktionalität (CPU) und Busklemmen

Die SPS-Funktionalität (CPU) des PFCs hat über absolute Adressen direkten Zugriff auf die Busklemmendaten.

Der PFC spricht die Eingangsdaten mit absoluten Adressen an. Die Daten können dann controller-intern über das IEC-61131-3-Programm verarbeitet werden. Merker werden dabei in einem remanenten Speicherbereich abgelegt. Anschließend können die Verknüpfungsergebnisse direkt über die absolute Adressierung in die Ausgangsdaten geschrieben werden.

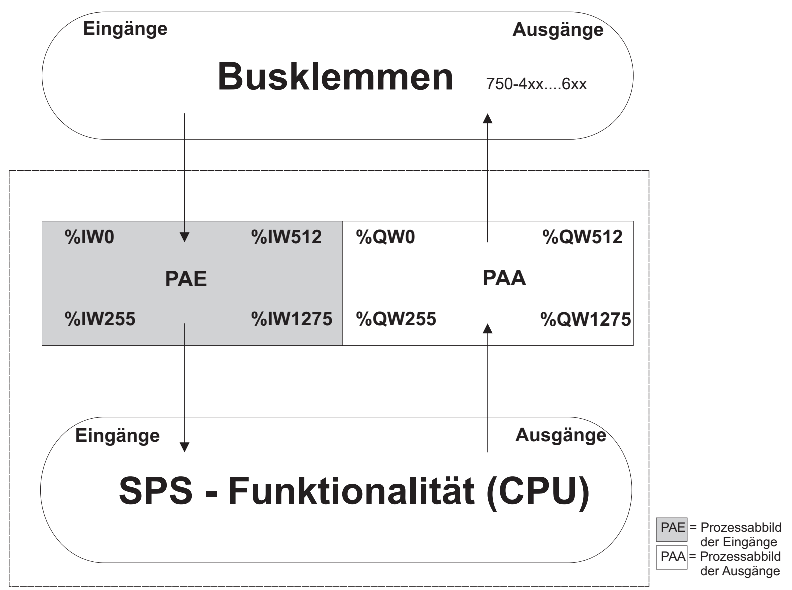


Abbildung 42: Datenaustausch zwischen SPS-Funktionalität (CPU) des PFCs und Busklemmen

7.3.5 Datenaustausch Master und SPS-Funktionalität (CPU)

Der Feldbusmaster und die SPS-Funktionalität (CPU) des Feldbuscontrollers haben unterschiedliche Sichtweisen auf die Daten.



Vom Master erzeugte Variablendaten gelangen als Eingangsvariablen zum Feldbuscontroller und werden dort weiter bearbeitet.
In dem Feldbuscontroller erstellte Daten werden als Ausgangsvariablen über den Feldbus zum Master gesendet.

In dem Feldbuscontroller kann ab Wortadresse 256 bis 511 (Doppelwortadresse 128-255, Byteadresse 512-1023) auf die MODBUS/TCP PFC-Variablendaten zugegriffen werden und ab Wortadresse 1276 bis 1531 (Doppelwortadresse 638-765, Byteadresse 2552-3063) auf die Variablendaten des Feldbuscontrollers.

7.3.5.1 Beispiel MODBUS/TCP-Master und SPS-Funktionalität (CPU)

Datenzugriff vom MODBUS/TCP-Master

Von dem MODBUS-Master wird grundsätzlich wortweise oder bitweise auf die Daten zugegriffen.
Die Adressierung der ersten 256 Datenworte von den Busklemmen beginnt beim wortweisen und bitweisen Zugriff bei 0.
Die Adressierung der Daten von den Variablen beginnt beim wortweisen Zugriff bei Wort 256, beim bitweisen Zugriff erfolgt die Adressierung dann ab:

4096 für Bit 0 im Wort 256
4097 für Bit 1 im Wort 256
...
8191 für Bit 15 im Wort 511.

Die Bit-Nummer lässt sich mit folgender Formel bestimmen:

BitNr = (Wort * 16) + Bitnr_im_Wort
Beispiel: 4097 = (256 * 16) + 1

Datenzugriff von der SPS-Funktionalität (CPU)

Die SPS-Funktionalität des PFCs verwendet bei dem Zugriff auf dieselben Daten eine andere Art der Adressierung. Bei der Deklaration von 16Bit-Variablen ist die SPS-Adressierung identisch mit der wortweisen Adressierung des MODBUS-Masters. Bei der Deklaration von booleschen Variablen (1 Bit) wird im Gegensatz zum MODBUS eine andere Notation verwendet. Hierbei setzt sich die Bitadresse aus den Elementen Wortadresse und Bitnummer im Wort zusammen, getrennt durch einen Punkt.

Beispiel:

Bitzugriff MODBUS auf Bitnummer 4097 => Bitadressierung in der SPS
<Wortnr>.<Bitnr> = 256.1

Die SPS-Funktionalität des PFCs kann außerdem byteweise und doppelwortweise auf die Daten zugreifen.

Bei dem byteweisen Zugriff errechnen sich die Adressen nach folgenden Formeln:

High-Byte Adresse = Wortadresse*2
Low-Byte Adresse = (Wortadresse*2) + 1

Bei dem doppelwortweisen Zugriff errechnet sich die Adresse nach folgender Formel:

Doppelwort Adresse = High-Wortadresse/2 (abgerundet)
oder = Low-Wortadresse/2

Information



Weitere Information
Eine detaillierte Beschreibung der MODBUS- und der entsprechenden IEC61131-Adressierung finden Sie in dem Kapitel „MODBUS-Register-Mapping“.

7.3.6 Anwendungsbeispiel

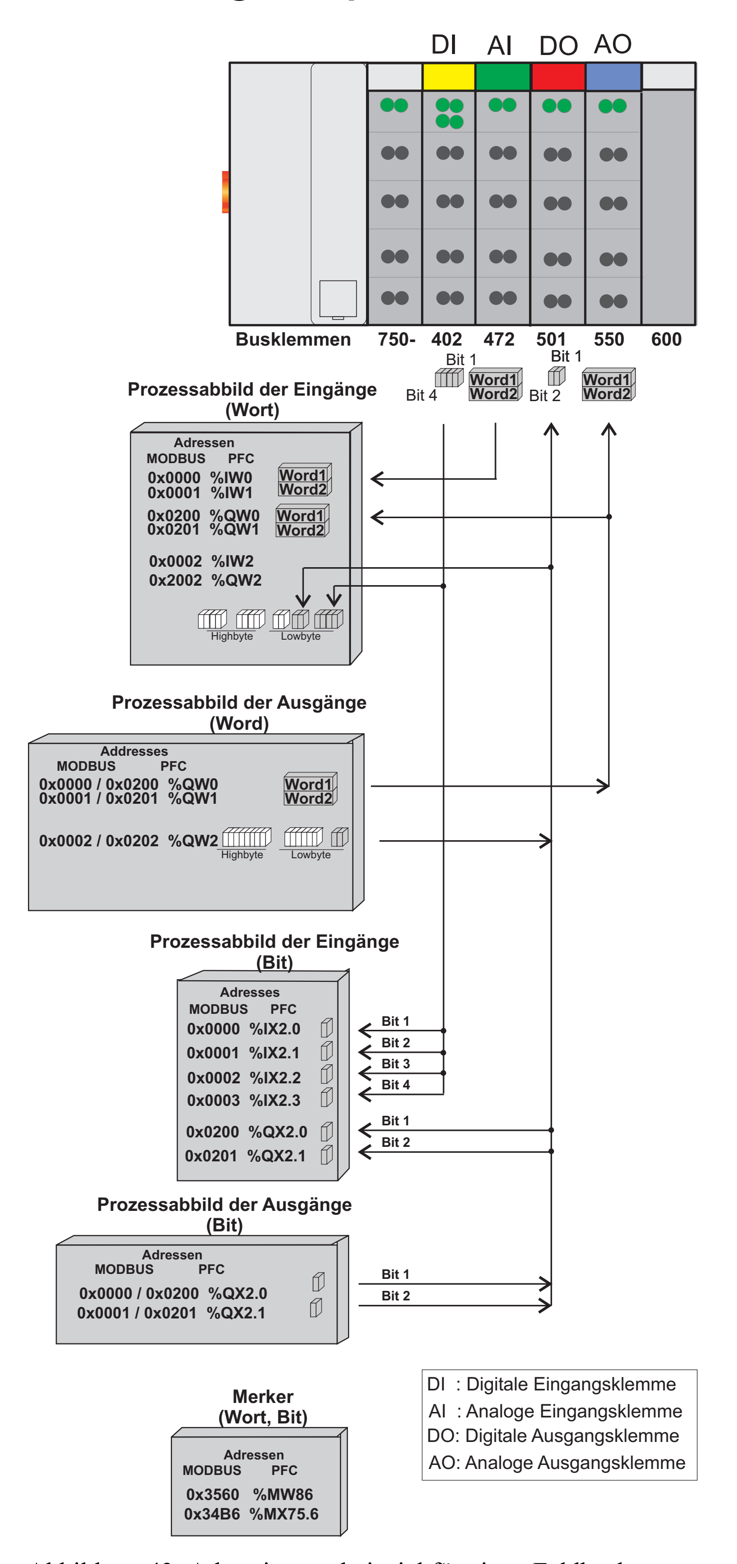


Abbildung 43: Adressierungsbeispiel für einen Feldbusknoten

8 In Betrieb nehmen

In diesem Kapitel wird Ihnen exemplarisch die Vorgehensweise für die Inbetriebnahme eines Feldbusknotens schrittweise aufgezeigt.

Hinweis



Exemplarisches Beispiel!
Diese Beschreibung ist exemplarisch und beschränkt sich hier auf die Ausführung einer lokalen Inbetriebnahme eines einzelnen Feldbusknoten mit einem nicht vernetzten Rechner unter Windows.

Für die Inbetriebnahme sind drei Arbeitsschritte erforderlich. Die Beschreibung dieser Arbeitsschritte finden Sie in den entsprechenden nachfolgenden Kapiteln.

- **Client-PC und Feldbusknoten anschließen**
- **IP-Adresse an den Feldbusknoten vergeben**
- **Funktion des Feldbusknotens testen**



Um Zugriff auf die controller-internen Konfigurationsseiten (Web-based Management-System) zu erhalten, ist die Zuweisung einer IP-Adresse an den Feldbuscontroller notwendig.

Hinweis



Die IP-Adresse muss im Netzwerk einmalig sein!
Für eine fehlerfreie Netzwerkkommunikation, beachten Sie, dass die zugewiesene IP-Adresse im Netzwerk einmalig sein muss!
Im Fehlerfall wird Ihnen beim nächsten Neustart über die I/O-LED die Fehlermeldung „Fehler in der IP-Adresskonfiguration“ (Fehlercode 6 - Fehlerargument 6) angezeigt.

Es gibt verschiedene Möglichkeiten, die IP-Adresse zu vergeben. Diese werden in den nachfolgenden Kapiteln einzeln beschrieben.

Im Anschluss an die Inbetriebnahmekapitel zur Vorbereitung der Kommunikation werden zusätzlich die folgenden Themen beschrieben:

- **Flash-Dateisystem vorbereiten**
- **Echtzeituhr synchronisieren**
- **Werkseinstellungen wiederherstellen**

Nach den oben genannten Themen finden Sie Hinweise zur Programmierung des Feldbuscontrollers mit WAGO-I/O-*PRO* CAA und die Beschreibung der internen Webseiten des Web-based Management-Systems (WBM) für weitere Einstellungen des Feldbuscontrollers.

8.1 PC und Feldbusknoten anschließen

1. Schließen Sie den Feldbusanschluss Ihres mechanisch und elektrisch montierten Feldbusknotens mit dem Feldbuskabel an eine entsprechende freie Schnittstelle Ihres Computers an.
   Der PC muss für diesen Anschluss über eine Netzwerkkarte verfügen.
   Die Übertragungsrate ist dann abhängig von der Netzwerkkarte Ihres PCs.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot.
Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, wird dieser mittels der I/O-LED durch rotes Blinken als Fehlercode ausgegeben.
Wird nach Anlauf des Feldbuskopplers über die I/O-LED durch 6-maliges rotes Blinken der Fehlercode 6 und anschließend durch 4-maliges rotes Blinken das Fehlerargument 4 ausgegeben, zeigt dieses an, dass noch keine IP-Adresse zugewiesen wurde.

8.2 IP-Adresse an den Feldbusknoten vergeben

- **IP-Adresse mit WAGO-ETHERNET-Settings vergeben**
  (über die serielle Kommunikationsschnittstelle)
- **IP-Adresse mit dem WAGO-BootP-Sever vergeben**
  (statisch über den Feldbus, wobei im Vergleich zur IP-Vergabe mit WAGO-ETHERNET-Settings mehrere Schritte notwendig sind)

8.2.1 IP-Adresse mit WAGO-ETHERNET-Settings vergeben

Die Windows-Anwendung „WAGO-ETHERNET-Settings 759-316“ ist eine Software, mit welcher busspezifische Parameter der ETHERNET-basierten WAGO-Feldbuskoppler und -controller gelesen und geändert werden können.

Mit den WAGO-ETHERNET-Settings können Sie folgende Funktionen nutzen:
- bei der Inbetriebnahme über die serielle Kommunikationsschnittstelle eine IP-Adresse konfigurieren
- Feldbuskoppler- und Felbuscontrollerparameter auf Werkseinstellungen zurücksetzen
- das Flash-Dateisystem, auf dem die HTML-Seiten des Feldbuskopplers/-controllers abgelegt sind, löschen und entpacken

Information



Weitere Informationen zu den WAGO-ETHERNET-Settings
Sie erhalten das Programm „WAGO-ETHERNET-Settings“ zum Herunterladen auf der Internetseite http://www.wago.com unter Downloads → AUTOMATION.



Das Programm ist außerdem auf der DVD-ROM „AUTOMATION Tools and Docs“ (Art.-Nr.: 0888-0412) zu beziehen.
Eine Kurzbeschreibung hierzu können Sie der „Schnellstartanleitung“ zum ETHERNET-Feldbuscontroller 750-841 entnehmen. Diese finden Sie ebenfalls auf der CD und auf den WAGO-Internetseiten unter http://www.wago.com unter Service → Downloads → Dokumentation.

Zur Datenkommunikation können WAGO-Kommunikationskabel oder WAGO-Funkadapter verwendet werden.

ACHTUNG



Kommunikationskabel 750-920 nicht unter Spannung stecken!
Um Schäden an der Kommunikationsschnittstelle zu vermeiden, stecken und ziehen Sie das Kommunikationskabel 750-920 nicht unter Spannung!
Der Feldbuscontroller muss dazu spannungsfrei sein!

1. Schalten Sie die Betriebsspannung des Feldbuscontrollers aus.

2. Schließen Sie das Kommunikationskabel 750-920 an die Konfigurationsschnittstelle des Feldbuscontrollers und an eine serielle Schnittstelle Ihres PCs an.

3. Schalten Sie die Betriebsspannung des Feldbuscontrollers wieder ein.

4. Starten Sie das Programm **WAGO-ETHERNET-Settings**.

5. Klicken Sie auf **[Read]**, um den angeschlossenen Feldbusknoten einzulesen und zu identifizieren.

6. Wählen Sie das Register **TCP/IP**:



Abbildung 44: Adresseinstellung in WAGO-ETHERNET-Settings

7. Damit Sie eine feste Adresse vergeben können, wählen Sie die Option für die Adressvergabe „Use following addresses“ aus (standardmäßig ist BootP aktiviert).

8. Geben Sie die gewünschte **IP-Adresse** und gegebenenfalls die Adresse der Subnetzmaske und des Gateways ein.

9. Klicken Sie auf die Schaltfläche **[Write]**, um die Adresse in den Feldbusknoten zu übernehmen.

10. Sie können nun WAGO-ETHERNET-Settings schließen oder bei Bedarf direkt im Web-based Management-System weitere Einstellungen vornehmen. Klicken Sie dazu auf **[Call WBM]**.

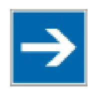
8.2.2 IP-Adresse mit dem WAGO-BootP-Server vergeben

Mittels „WAGO-BootP-Server“ oder SPS-Programm kann eine feste IP-Adresse vergeben werden.

Die Vergabe mittels SPS-Programm wird durch den Funktionsblock „Ethernet_Set_Network_Config“ der Bibliothek „Ethernet.lib“ realisiert, welcher in WAGO-I/O-*PRO* CAA eingebunden wird.

Im Folgenden wird die IP-Adressvergabe für den Feldbusknoten exemplarisch mittels des WAGO-BootP-Servers beschrieben.

Hinweis



IP-Adressvergabe nicht über Router möglich!
Die Vergabe der IP-Adresse erfolgt über ein Straight-Through-Kabel, Switches, Hubs oder in einer direkten Verbindung mittels eines Cross-over-Kabels. Über einen Router ist keine Adressenvergabe möglich.

Hinweis



BootP muss im Web-based Management-System aktiviert sein!
Beachten Sie, dass BootP auf den internen Webseiten des WBM, auf der HTML-Seite „Port“, aktiviert sein muss.
Im Auslieferungszustand ist BootP standardmäßig aktiviert.

Information



Weitere Information
Die IP-Adressvergabe mittels WAGO-BootP-Server ist unter Windows- und Linux-Betriebssystemen möglich. Neben dem WAGO-BootP-Server können beliebigen anderen BootP-Server verwendet werden.

Information



Weitere Informationen zu dem WAGO-BootP-Server
Sie erhalten den „WAGO-BootP-Server 759-315“ kostenlos auf der DVD-ROM „AUTOMATION Tools and Docs“ (Art.-Nr.: 0888-0412) oder auf der Internetseite http://www.wago.com unter Downloads → AUTOMATION → 759-315 WAGO-BootP-Server.

Die Beschreibung umfasst die folgenden Arbeitsschritte:

- MAC-ID ermitteln
- IP-Adresse ermitteln
- BootP-Tabelle editieren
- BootP aktivieren
- BootP deaktivieren

8.2.2.1 MAC-ID ermitteln

1. Notieren Sie die MAC-ID Ihres Feldbuscontrollers, bevor Sie Ihren Feldbusknoten aufbauen.
   Ist der Feldbuscontroller bereits verbaut, schalten Sie die Betriebsspannung des Feldbuscontrollers aus und nehmen Sie ihn aus dem Verbund heraus.

Die MAC-ID ist auf der Rückseite des Feldbuscontrollers oder auf dem selbstklebenden Abreißetikett seitlich auf dem Feldbuscontroller aufgebracht.

MAC-ID des Feldbuscontrollers: 0 0 : 3 0 : D E : _ _ : _ _ : _ _

2. Stecken Sie den Feldbuscontroller in den Verbund des Feldbusknotens.

3. Schließen Sie den Feldbusanschluss Ihres mechanisch und elektrisch montierten Feldbusknotens mit dem Feldbuskabel an eine entsprechende freie Schnittstelle Ihres Computers an.
   Der Client-PC muss für diesen Anschluss über eine Netzwerkkarte verfügen.
   Die Übertragungsrate ist dann abhängig von der Netzwerkkarte Ihres Client-PCs.

Hinweis



Bei direkter Verbindung zum PC ein Cross-over-Kabel verwenden!
Erfolgt die Verbindung direkt mit einem Client-PC, wird statt eines Straight-Through-Kabels (1:1) ein sogenanntes Cross-over-Kabel benötigt.

4. Starten Sie den Client-PC, der die Funktion des Masters und BootP-Servers übernimmt.

5. Schalten Sie die Spannungsversorgung am Controller (DC-24V-Netzteil) ein.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot.
Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, der mittels I/O-LED durch rotes Blinken als Fehlermeldung ausgegeben wird, werten Sie Fehlercode und -argument aus und beheben Sie den Fehler.

Information



Weitere Informationen zu der LED-Signalisierung
Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose“, „LED-Signalisierung“.

Wird nach Anlauf des Feldbuscontrollers durch 6-maliges rotes Blinken der Fehlercode 6 und anschließend durch 4-maliges rotes Blinken das Fehlerargument



4 mittels I/O-LED ausgegeben, zeigt dies an, dass noch keine IP-Adresse zugewiesen wurde.

8.2.2.2 IP-Adresse ermitteln

1. Ist der Client-PC bereits in ein IP-Netzwerk eingebunden, können Sie die IP-Adresse des Client-PCs ermitteln, indem Sie auf Ihrer Bildschirmoberfläche über das **Startmenü / Einstellungen** gehen und auf **Systemsteuerung** klicken.

2. Klicken Sie doppelt auf das Icon **Netzwerk**
   Das Netzwerk-Dialogfenster wird geöffnet.

   Unter Windows NT:

   - Wählen Sie das Register **Protokolle**
   - Markieren Sie den Eintrag TCP/IP Protokoll

   Unter Windows 2000/XP:

   - Wählen Sie Netzwerk- und DFÜ-Verbindungen
   - In dem sich öffnenden Fenster klicken Sie mit der rechten Maustaste auf **LAN-Verbindung** und öffnen die Eigenschaften der Verbindung.
   - Markieren Sie den Eintrag **Internetprotokoll TCP/IP**

Hinweis



TCP/IP-Komponente bei Bedarf nachinstallieren!
Fehlt der Eintrag „Internetprotokoll TCP/IP“, installieren Sie die entsprechende TCP/IP-Komponente, und starten Sie Ihren PC neu.
Für die Installation benötigen Sie die Installations-CD für Windows NT, 2000 oder XP.

3. Klicken Sie anschließend auf die Schaltfläche **Eigenschaften...**

4. In dem Eigenschaftenfenster entnehmen Sie die IP-Adresse, die Subnetzmaske und gegebenenfalls die Adresse für das Gateway Ihres Client-PCs, und notieren Sie diese Werte:

   IP-Adresse Client-PC: _ _ _ . _ _ _ . _ _ _ . _ _ _
   Subnetzmaske: _ _ _ . _ _ _ . _ _ _ . _ _ _
   Gateway: _ _ _ . _ _ _ . _ _ _ . _ _ _

5. Wählen Sie nun eine gewünschte IP-Adresse für Ihren Feldbusknoten.

Hinweis



Client-PC feste IP-Adresse zuweisen und auf gemeinsames Subnetz achten!
Beachten Sie, dass der Client-PC, auf dem der BootP-Server ausgeführt wird, eine feste IP-Adresse haben muss, und dass der Feldbusknoten und der Client-PC sich in demselben Subnetz befinden müssen.

6. Notieren Sie sich die von Ihnen gewählte IP-Adresse:

   IP-Adresse Feldbusknoten: ___ . ___ . ___ . ___

8.2.2.3 BootP-Tabelle editieren

Die BootP-Tabelle stellt die Datenbasis für den BootP-Server dar. Sie liegt in Form einer Textdatei (bootptab.txt) auf dem Client-PC, auf welchem der WAGO-BootP-Server installiert ist.

Hinweis



Für weitere Konfiguration BootP-Server installieren!
Voraussetzung für die folgenden Schritte ist die korrekte Installation des WAGO-BootP-Servers.

1. Öffnen Sie auf Ihrem PC das **Startmenü** und wählen Sie den Menüpunkt **Programme \ WAGO Software \ WAGO BootP Server** aus.

2. Klicken Sie auf **WAGO BootP Server Konfiguration**.

Sie erhalten die editierbare Tabelle „bootptab.txt“.

Im Anschluss an die Auflistung aller Kürzel, die in der BootP-Tabelle verwendet werden können, sind am Ende der Tabelle zwei Beispiele für die Vergabe einer IP-Adresse aufgeführt:

- **„Example of entry with no gateway“**
- **„Example of entry with gateway“**

Für das in dieser Beschreibung behandelte lokale Netzwerk wird kein Gateway benötigt. Somit können Sie das Beispiel „**Example of entry with no gateway**“ verwenden.

bootptab.txt - Editor

Datei Bearbeiten Suchen ?

```
# sequence of bytes where each byte is a two-digit hex value.
#
# Example of entry with no gateway
node1:ht=1:ha=0030DE000100:ip=10.1.254.100
#
# Example of entry with gateway
node2:ht=1:ha=0030DE000200:ip=10.1.254.200:T3=0A.01.FE.01
```

Abbildung 45: BootP-Tabelle

Die aufgeführten Beispiele enthaltenen folgende Informationen:



Tabelle 38: Informationen der BootP-Tabelle

Angabe	Bedeutung
node1, node2	Hier kann ein beliebiger Name für den Knoten vergeben werden.
ht=1	Hier wird der Hardware-Typ des Netzwerkes angegeben. Für ETHERNET gilt der Hardware-Typ 1. Die Nummern sind im RFC1700 beschrieben.
ha=0030DE000100 ha=0030DE000200	Hier wird die Hardware-Adresse (MAC-ID) des Feldbuscontrollers angegeben (hexadezimal).
ip= 10.1.254.100 ip= 10.1.254.200	Hier wird die IP-Adresse des Feldbuscontrollers angegeben (dezimal).
T3=0A.01.FE.01	Hier wird die Gateway-Adresse angegeben (hexadezimal).
sm=255.255.0.0	Zusätzlich kann die Subnetzmaske des Subnetzes eingetragen werden, zu dem der Feldbuscontroller gehört (dezimal).

Für das in dieser Beschreibung behandelte lokale Netzwerk wird kein Gateway benötigt.
Somit kann das obere Beispiel „Example of entry with no gateway“ verwendet werden.

3. Tauschen Sie in folgender Textzeile die zwölfstellige Hardware-Adresse aus, die in dem Beispiel hinter „ha=“ eingetragen ist.

   node1:ht=1:ha=0030DE000100:ip=10.1.254.100

4. Geben Sie an dieser Stelle die MAC-ID Ihres eigenen Feldbuscontrollers ein.

5. Wenn Sie Ihrem Feldbusknoten einen Namen geben möchten, löschen Sie den Namen „node1“ und tragen Sie an dieser Stelle einen beliebigen Namen ein.

   node1:ht=1:ha=0030DE000100:ip=10.1.254.100

6. Um dem Feldbuscontroller eine gewünschte IP-Adresse zuzuweisen, markieren Sie die in dem Beispiel angegebene IP-Adresse, die hinter „ip=“ eingetragen ist und geben die von Ihnen gewählte IP-Adresse ein.

   node1:ht=1:ha=0030DE000100:ip=10.1.254.100

7. Da Sie das zweite Beispiel „**Example of entry with gateway**“ an dieser Stelle nicht benötigen, setzen Sie als Kommentar-Zeichen die Raute (#) vor die Textzeile von Beispiel 2:

   # node2:hat=1:ha=003 0DE 0002 00:ip=10.1.254.200:T3=0A.01.FE.01

   Diese Zeile wird nachfolgend nicht mehr ausgewertet.

Hinweis



Adressen weiterer Knoten in bootptap.txt eintragen!
Für die Adressierung weiterer Feldbusknoten geben Sie für jeden Knoten analog eine entsprechende Textzeile mit den gewünschten Einträgen ein.

8. Wählen Sie im Menü **Datei** den Menüpunkt **Speichern** aus, um die geänderten Einstellungen in der Datei „bootptab.txt“ zu sichern.

9. Schließen Sie den Editor.

8.2.2.4 BootP aktivieren

1. Öffnen Sie auf Ihrem PC das **Startmenü** und wählen Sie den Menüpunkt **Programme \ WAGO Software \ WAGO BootP Server** aus.

2. Klicken Sie auf **WAGO BootP Server** um das Dialogfenster zu öffnen.

3. Klicken Sie in dem sich öffnenden Dialogfenster auf die Schaltfläche **[Start]**. Dadurch wird der Frage-Antwort-Mechanismus des BootP-Protokolls aktiviert.

In dem BootP-Server werden nun eine Reihe von Meldungen ausgegeben. Die Fehlermeldungen zeigen Ihnen an, dass in dem Betriebssystem einige Services (z. B. Port 67, Port 68) nicht definiert sind.

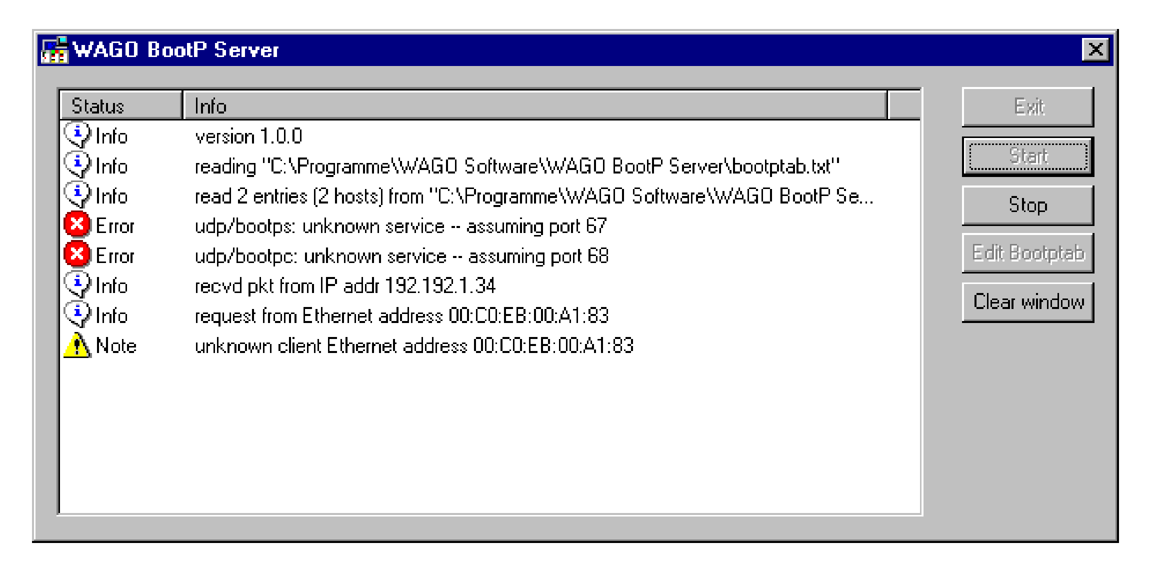


Abbildung 46: Dialogfenster des WAGO-BootP-Servers mit Meldungen

Damit die neue IP-Adresse in den Feldbuscontroller übernommen wird, starten Sie den Feldbuscontroller zu diesem Zeitpunkt unbedingt durch einen Hardware-Reset neu:

4. Unterbrechen Sie die Spannungsversorgung des Feldbuscontrollers für ca. 2 Sekunden oder drücken Sie den Betriebsartenschalters herunter, der sich hinter der Konfigurationsschnittstellen-Klappe befindet.

Die IP-Adresse ist fest im Feldbuscontroller gespeichert.

5. Um den BootP-Server wieder zu schließen, klicken Sie auf die Schaltfläche **[Stop]** und dann auf die Schaltfläche **[Exit]**.

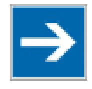


8.2.2.5 BootP deaktivieren

Bei aktiviertem BootP-Protokoll erwartet der Controller die permanente Anwesenheit eines BootP-Servers. Ist jedoch nach einem Power-On-Reset kein BootP-Server verfügbar, dann bleibt das Netzwerk inaktiv.

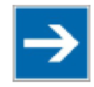
Sie müssen das BootP-Protokoll deaktivieren, damit der Controller die konfigurierte IP-Adresse aus dem EEPROM verwendet, so ist keine Anwesenheit eines BootP-Servers mehr erforderlich.

Hinweis



Für die dauerhafte Adressvergabe, muss BootP deaktiviert werden!
Damit die neue IP-Adresse dauerhaft in den Feldbuscontroller übernommen wird, müssen Sie BootP deaktivieren.
Damit wird ausgeschlossen, dass der Feldbuscontroller eine erneute BootP-Anfrage erhält.

Hinweis



Kein Verlust der IP-Adresse bei deaktiviertem BootP-Protokoll!
Ist das BootP-Protokoll nach der Adressvergabe deaktiviert, bleibt die gespeicherte IP-Adresse auch erhalten, wenn es einen längeren Spannungsausfall gibt oder der Controller ausgebaut wird.

Das Deaktivieren von BootP können Sie auf zwei Arten vornehmen:

- BootP in **WAGO-ETHERNET-Settings** deaktivieren
- BootP im **Web-based Management-System** deaktivieren

BootP in WAGO-ETHERNET-Settings deaktivieren

ACHTUNG



Kommunikationskabel 750-920 nicht unter Spannung stecken!
Um Schäden an der Kommunikationsschnittstelle zu vermeiden, stecken und ziehen Sie das Kommunikationskabel 750-920 nicht unter Spannung!
Der Feldbuscontroller muss dazu spannungsfrei sein!

1. Schalten Sie die Betriebsspannung des Feldbuscontrollers aus.

2. Schließen Sie das Kommunikationskabel 750-920 an die Konfigurationsschnittstelle des Feldbuscontrollers und an eine serielle Schnittstelle Ihres PCs an.

3. Schalten Sie die Betriebsspannung des Feldbuscontrollers wieder ein.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot.
Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, der mittels I/O-LED durch rotes Blinken als Fehlermeldung ausgegeben wird, werten Sie Fehlercode und -argument aus und beheben Sie den Fehler.

Information



Weitere Informationen zu der LED-Signalisierung
Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose", „LED-Signalisierung".

Wird nach Anlauf des Feldbuscontrollers durch 6-maliges rotes Blinken der Fehlercode 6 und anschließend durch 4-maliges rotes Blinken das Fehlerargument 4 mittels I/O-LED ausgegeben, zeigt dies an, dass noch keine IP-Adresse zugewiesen wurde.

4. Starten Sie das Programm **WAGO-ETHERNET-Settings**.

5. Klicken Sie auf **[Read]**, um den angeschlossenen Feldbusknoten einzulesen und zu identifizieren.

6. Wählen Sie das Register **TCP/IP**:

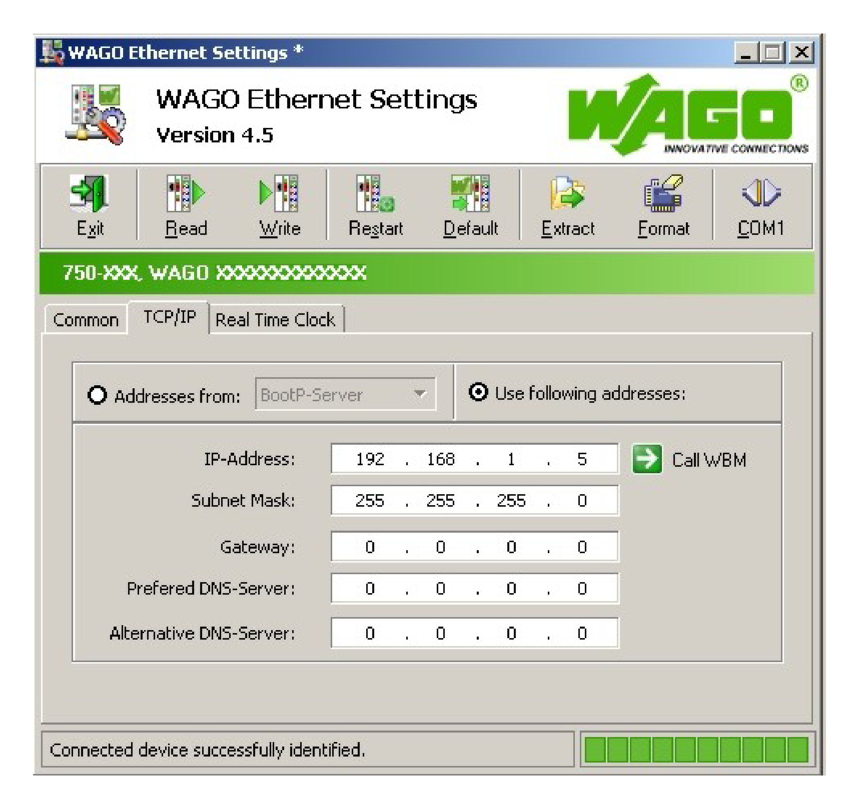


Abbildung 47: Adresseinstellung in WAGO-ETHERNET-Settings

7. Damit Sie eine feste Adresse vergeben können, wählen Sie die Option für die Adressvergabe „Use following addresses" aus (standardmäßig ist BootP aktiviert).

8. Klicken Sie auf die Schaltfläche **[Write]**, um die Adresse in den Feldbusknoten zu übernehmen.



9. Sie können nun WAGO-ETHERNET-Settings schließen oder bei Bedarf direkt im Web-based Management-System weitere Einstellungen vornehmen. Klicken Sie dazu auf **[Call WBM]**.

BootP in dem Web-based Management-System deaktivieren

1. Öffnen Sie auf Ihrem Client-PC einen **Web-Browser** (z. B. Microsoft Internet Explorer) für die Anzeige der Feldbuscontroller-internen HTML-Seiten (Web-based Management-System).

2. Geben Sie die **IP-Adresse** Ihres Feldbusknotens in das Adressfeld des Browsers ein und drücken Sie die Taste **[Enter]**.

Sie erhalten ein Dialogfenster mit einer Passwort-Abfrage. Diese dient der Zugriffssicherung und enthält die drei verschiedenen Benutzergruppen: „admin", „guest" und „user".

3. Geben Sie als Administrator den Benutzernamen **„admin"** und das Kennwort **„wago"** ein.

In dem Browser-Fenster wird eine Startseite mit den Informationen zu Ihrem Feldbuscontroller angezeigt (Startseite auf Seite „PLC" änderbar).
Über Hyperlinks in der linken Navigationsleiste gelangen Sie zu den weiteren Informationen.

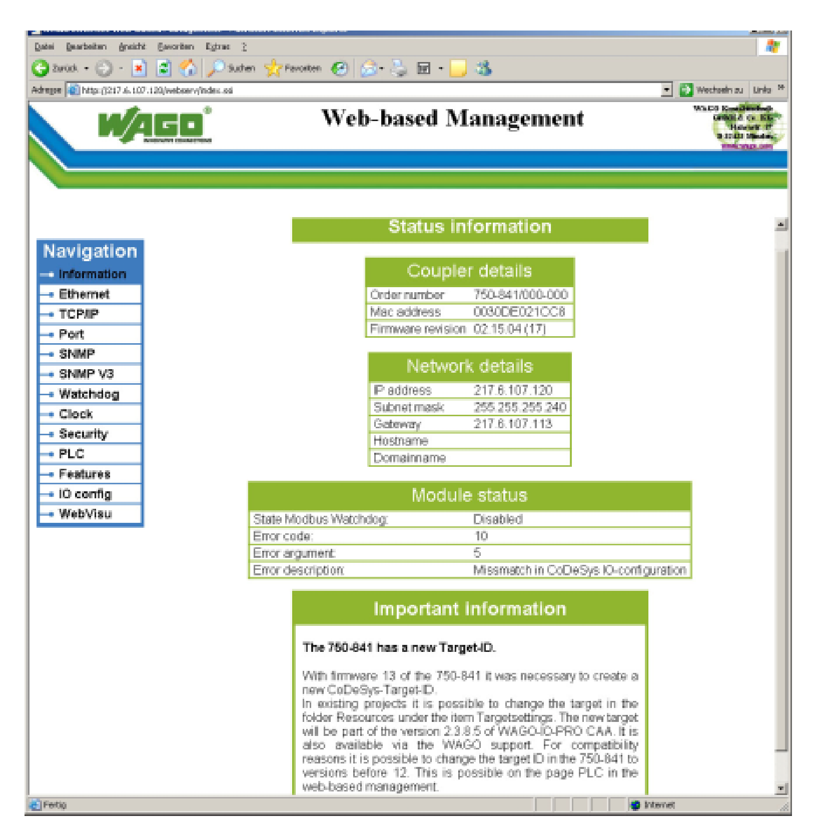
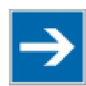


Abbildung 48: WBM-Seite „Information"

Hinweis



Zur Anzeige des Web-based Management-Systems Proxy-Server deaktivieren!
Werden bei dem lokalen Zugriff auf den Feldbusknoten die Seiten nicht angezeigt, dann definieren Sie bitte in Ihrem Web-Browser, dass für die IP-Adresse des Knoten ausnahmsweise kein Proxy-Server verwendet werden soll.

Hinweis



Änderung der Controller-IP durch DHCP-Server im Netz möglich!
Sind DHCP und BootP aktiviert und wird in das Netzwerk ein DHCP-Server oder ein Router mit aktivem DHCP-Server eingebaut, erfolgt nach einem Netzausfall (Ausfall der 24-V-DC-Versorgung der Controller) eine automatische Adressvergabe aus dem Adressbereich des DHCP-Servers. Das heißt, alle Feldbuskoppler/-controller erhalten eine neue IP-Adresse!

4. Klicken Sie in der linken Navigationsleiste auf den Link **Port**, um die HTML-Seite für die Protokollauswahl zu öffnen.

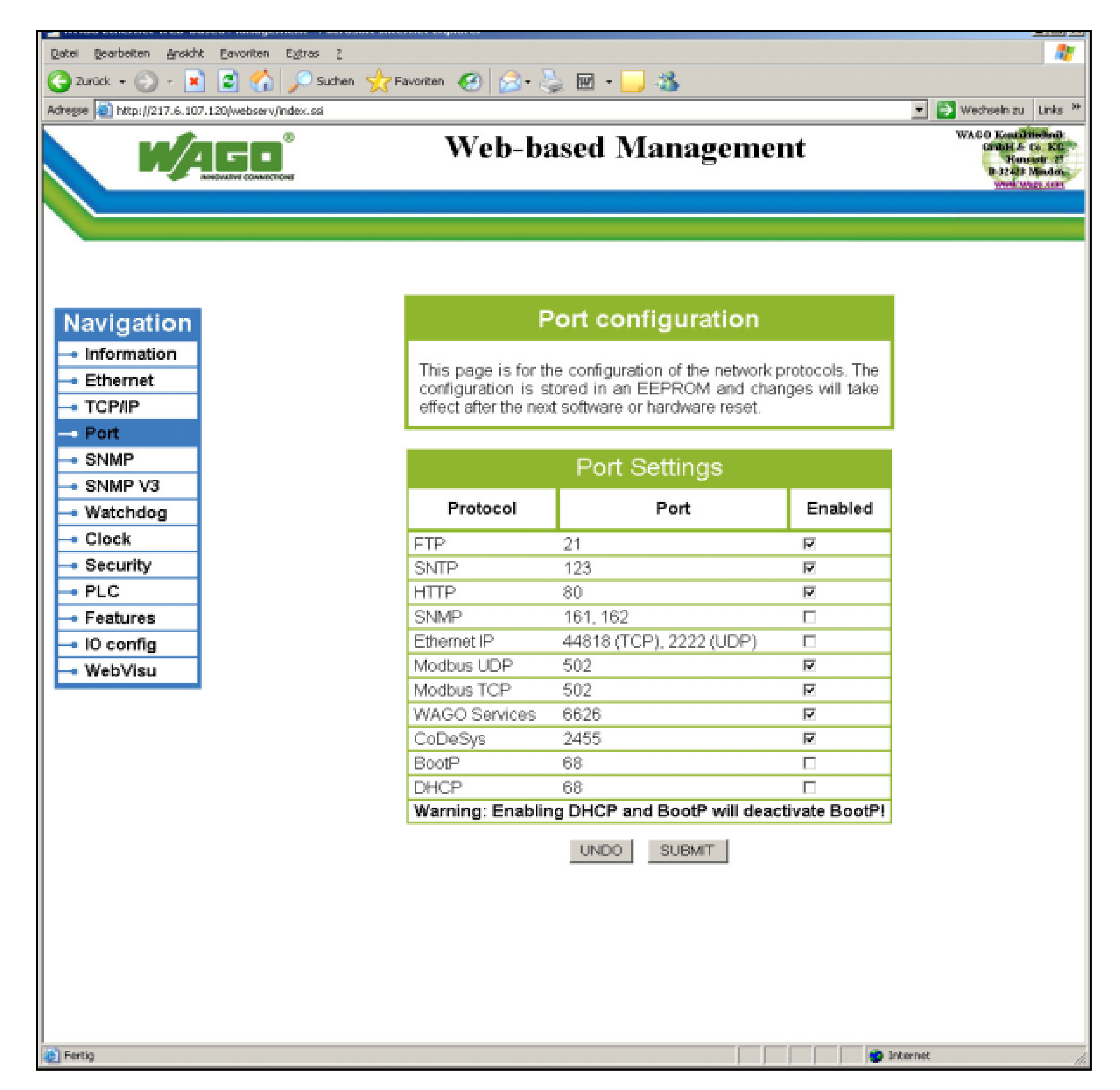


Abbildung 49: WBM-Seite „Port“

Sie erhalten eine Liste aller Protokolle, die der Controller unterstützt.



5. Wählen Sie die Option „DHCP“ oder „use IP from EEPROM“.
   Sie haben das BootP Protokoll jetzt deaktiviert.

Darüber hinaus können Sie nun entsprechend weitere Protokolle deaktivieren, die Sie nicht benötigen oder Protokolle auswählen und explizit aktivieren, mit denen Sie arbeiten möchten.

Da die Kommunikation für jedes Protokoll über einen anderen Port stattfindet, können Sie mehrere Protokolle gleichzeitig aktivieren und über diese Protokolle kommunizieren.

6. Damit die vorgenommene Protokollauswahl übernommen wird, klicken Sie auf die Schaltfläche **SUBMIT** und unterbrechen Sie anschließend die Spannungsversorgung des Feldbuscontrollers (Hardware-Reset) oder drücken Sie den Betriebsartenschalter herunter.

Die Protokolleinstellungen sind gespeichert und der Controller ist betriebsbereit.

Haben Sie z. B. das MODBUS/TCP-Protokoll aktiviert, können Sie nun mit einem MODBUS-Mastertool gewünschte MODBUS-Funktionen auswählen und ausführen, so z. B. die Abfrage der Klemmenkonfiguration über das Register 0x2030.

Haben Sie beispielsweise WAGO-I/O-PRO aktiviert, können Sie den Controller auch über die ETHERNET-Verbindung mit WAGO-I/O-*PRO* CAA nach der Norm IEC-61131-3 programmieren.

8.2.2.6 Gründe für eine fehlgeschlagene IP-Adressvergabe

- Die MAC-Adresse des Feldbuscontrollers stimmt nicht mit dem Eintrag in der Datei „bootstrap.txt“ überein
- Der Client-PC auf dem der BootP-Server läuft, befindet sich nicht im gleichen Subnetz wie der Feldbuscontroller, d. h. die IP-Adressen passen nicht zusammen.
  Beispiel: Client-IP: 192.168.0.10 und Feldbuscontroller-IP: 10.1.254.5
- Client-PC und/oder Feldbuscontroller haben keine ETHERNET-Verbindung
- Die Signalqualität ist schlecht (Switches oder Hubs verwenden)

8.3 Funktion des Feldbusknotens testen

Information



Weitere Informationen zum Auslesen der IP-Adresse
Sie können mittels WAGO-ETHERNET-Settings die aktuell zugewiesene IP-Adresse auslesen.
Gehen Sie dazu, wie in dem Kapitel „IP-Adresse mit WAGO-ETHERNET-Settings vergeben“ beschrieben, vor.

1. Um die korrekte Vergabe der IP-Adresse und die Kommunikation mit dem Feldbusknoten zu testen, schalten Sie zunächst die Betriebsspannung des Feldbusknotens aus.

2. Stellen Sie eine nicht-serielle Feldbusverbindung zwischen Client-PC und Feldbusknoten her.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot.
Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, der mittels I/O-LED durch rotes Blinken als Fehlermeldung ausgegeben wird, werten Sie Fehlercode und -argument aus und beheben Sie den Fehler.

Information



Weitere Informationen zu der LED-Signalisierung
Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose“, „LED-Signalisierung“.

3. Rufen Sie die DOS-Eingabeaufforderung unter **Startmenü / Programme / Eingabeaufforderung** auf.

4. Geben Sie den Befehl **ping** mit der von Ihnen vergebenen IP-Adresse in der folgenden Schreibweise ein:

   ping [Leerzeichen] XXX . XXX . XXX . XXX

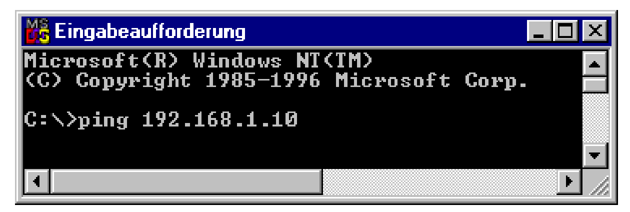


Abbildung 50: Beispiel für den Funktionstest eines Feldbusknotens

5. Drücken Sie die Taste **[Enter]**.
   Ihr Client-PC empfängt nun eine Antwort vom Feldbusknoten, die in der DOS-Eingabeaufforderung dargestellt wird.
   Falls stattdessen die Fehlermeldung „Zeitüberschreitung der Anforderung



(Timeout)“erscheint, vergleichen Sie Ihre Eingaben nochmals mit der zugewiesenen IP-Adresse.

6. Bei erfolgreichem Test schließen Sie die DOS-Eingabeaufforderung.

Der Feldbusknoten ist jetzt für die Kommunikation vorbereitet.



8.4 Flash-Dateisystem vorbereiten

Die Vorbereitung des Flash-Dateisystems ist erforderlich, damit Sie alle weiteren Konfigurationen über das Web-Interface des Feldbuscontrollers durchführen können.

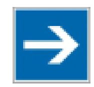
Im Auslieferungszustand ist das Flash-Dateisystem bereits vorbereitet.
Sollte jedoch bei Ihrem Feldbuscontroller das Flash-Dateisystem noch nicht initialisiert oder auf Grund eines Fehlers zerstört worden sein, müssen Sie dieses zunächst in den Flash-Speicher entpacken, damit Sie darauf zugreifen können.

ACHTUNG



Kommunikationskabel 750-920 nicht unter Spannung stecken!
Um Schäden an der Kommunikationsschnittstelle zu vermeiden, stecken und ziehen Sie das Kommunikationskabel 750-920 nicht unter Spannung!
Der Feldbuskoppler muss dazu spannungsfrei sein!

Hinweis



Daten werden durch Formatieren gelöscht!
Beachten Sie, dass durch das Formatieren des Dateisystems alle Daten und gespeicherte Konfigurationen gelöscht werden.
Verwenden Sie diese Funktion nur dann, wenn das Flash-Dateisystem noch nicht initialisiert oder auf Grund eines Fehlers zerstört wurde.

1. Schalten Sie die Betriebsspannung des Feldbuscontrollers aus.

2. Schließen Sie das Kommunikationskabel 750-920 an die Konfigurationsschnittstelle des Feldbuscontrollers und an eine serielle Schnittstelle Ihres PCs an.

3. Schalten Sie die Betriebsspannung des Feldbuscontrollers wieder ein.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot.
Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, der mittels I/O-LED durch rotes Blinken als Fehlermeldung ausgegeben wird, werten Sie Fehlercode und -argument aus und beheben Sie den Fehler.

Information



Weitere Informationen zu der LED-Signalisierung
Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose“, „LED-Signalisierung“.

4. Starten Sie das Programm **WAGO-ETHERNET-Settings.**



5. Um das Dateisystem zu formatieren, wählen Sie in der oberen Menüleiste die Schaltfläche **Format**.
   Das Formatieren ist beendet, sobald im unteren Statusfenster „Formatting flash disk successfully done“ angezeigt wird.

6. Wählen Sie in der oberen Menüleiste die Schaltfläche **Extract**, um die Web-Seiten in das Flash-Dateisystem zu extrahieren.
   Dieser Vorgang dauert ein paar Sekunden und ist beendet, sobald im Statusfenster „Extracting files successfully done“ angezeigt wird.

Hinweis



Neustart des Feldbuskopplers/-controllers nach Format/Extract!
Damit der Webseiten nach einem Format/Extract angezeigt werden können, muss der Feldbuskoppler/-controller neugestartet werden.

8.5 Echtzeituhr synchronisieren

Der Echtzeit-Uhrenbaustein des Feldbuscontrollers ermöglicht eine Datum- und Zeitangabe für Dateien im Flash-Dateisystem.
Synchronisieren Sie die Echtzeituhr bei der Inbetriebnahme mit der aktuellen Rechnerzeit.

Um die Echtzeituhr zu synchronisieren gibt es zwei Möglichkeiten:

- Echtzeituhr mit den **WAGO-ETHERNET-Settings** synchronisieren
- Echtzeituhr über das **Web-based Management-System** synchronisieren

Echtzeituhr mit den WAGO-ETHERNET-Settings synchronisieren

1. Schalten Sie die Betriebsspannung des Feldbuscontrollers aus.
2. Schließen Sie das Kommunikationskabel 750-920 an die Konfigurationsschnittstelle des Feldbuscontrollers und an eine serielle Schnittstelle Ihres PCs an.
3. Schalten Sie die Betriebsspannung des Feldbuscontrollers wieder ein.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot.
Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, der mittels I/O-LED durch rotes Blinken als Fehlermeldung ausgegeben wird, werten Sie Fehlercode und -argument aus und beheben Sie den Fehler.

Information



Weitere Informationen zu der LED-Signalisierung
Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose“, „LED-Signalisierung“.

4. Starten Sie das Programm **WAGO-ETHERNET-Settings.**
5. Wählen Sie das Register **Real Time Clock**.



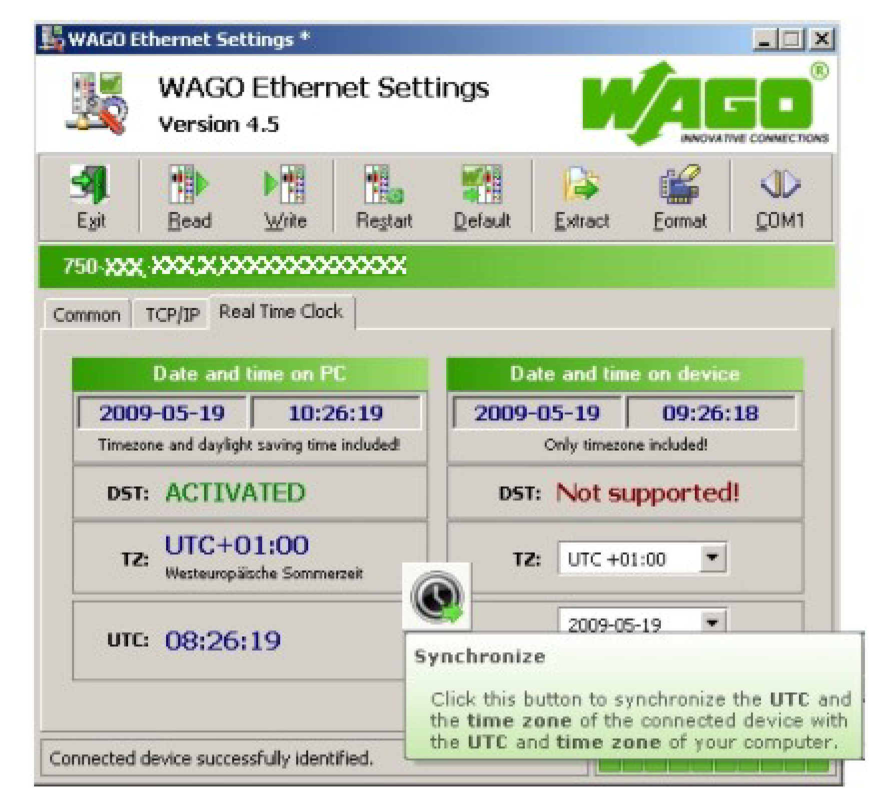


Abbildung 51: ETHERNET-Settings-Beispiel für die Echtzeituhrsynchronisation

6. Klicken Sie auf den Button mit dem Uhren-Symbol "Synchronize".

Echtzeituhr über das Web-based Management-System synchronisieren

1. Starten Sie einen Web-Browser (z.B. MS Internet-Explorer oder Mozilla) und geben Sie in der Adresszeile die IP-Adresse ein, die Sie Ihrem Feldbusknoten vergeben haben.

2. Bestätigen Sie mit **[Enter].**
   Die Startseite des Web-Interface wird aufgebaut.

3. Wählen Sie „Clock“ in der linken Menüleiste.

4. Geben Sie in der folgenden Abfrage Ihren Benutzernamen und das Passwort ein (Default: User = „admin“, Passwort = „wago“ oder: User = „user“, Passwort = „user“).
   Die HTML-Seite "Clock configuration" wird aufgebaut:

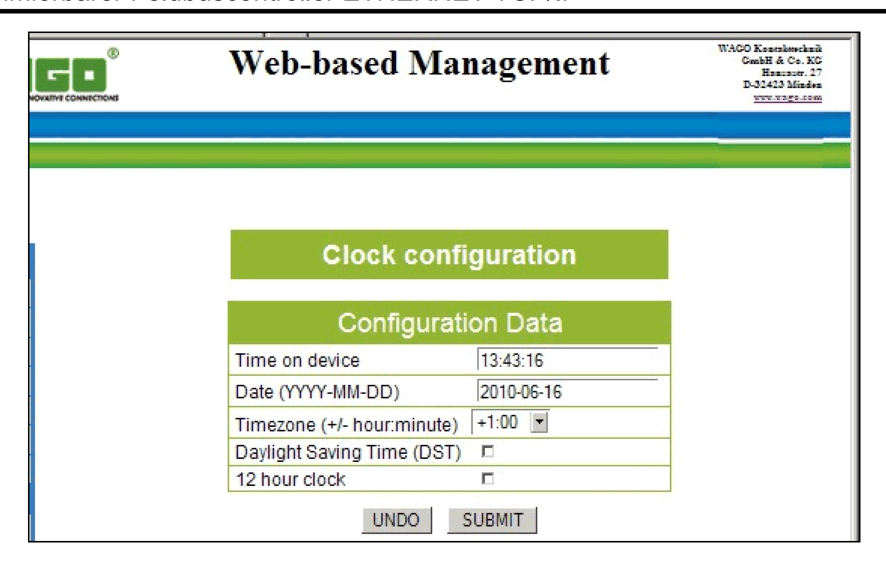


Abbildung 1: Beispiel WBM Clock configuration

5. Stellen Sie die Werte in den Feldern „Time on device“, „Date“ und „Timezone“ auf die entsprechend aktuellen Werte ein, und aktivieren Sie gegebenenfalls die Option "Daylight Saving Time (DST)".

6. Klicken Sie auf **[SUBMIT]**, um die Änderungen in Ihren Feldbusknoten zu übernehmen.

7. Damit die Einstellungen des Web-Interface wirksam werden, führen Sie einen Neustart des Feldbusknotens durch.



8.6 Werkseinstellungen wiederherstellen

Um die werksseitigen Einstellungen wiederherzustellen, gehen Sie wie folgt vor:

1. Schalten Sie die Betriebsspannung des Feldbuscontrollers aus.
2. Schließen Sie das Kommunikationskabel 750-920 an die Konfigurationsschnittstelle des Feldbuscontrollers und an eine serielle Schnittstelle Ihres PCs an.
3. Schalten Sie die Betriebsspannung des Feldbuscontrollers wieder ein.
4. Starten Sie das Programm **WAGO-ETHERNET-Settings.**
5. Wählen Sie in der oberen Menüleiste die Schaltfläche **Default** und bestätigen Sie die folgende Abfrage mit **[Yes]**.

Es wird automatisch ein Neustart des Feldbusknotens ausgeführt.
Der Start erfolgt mit den Werkseinstellungen.

9 PFC mit WAGO-I/O-*PRO* CAA programmieren

Durch die IEC-61131-3-Programmierung kann ein Programmierbarer Feldbuscontroller ETHERNET TCP/IP über die Funktionen eines Feldbuskopplers hinaus die Funktionalität einer SPS nutzen. Die Applikation gemäß IEC-61131-3 erstellen Sie mit dem Programmiertool WAGO-I/O-*PRO* CAA.

Hinweis



Option „CoDeSys“ im Web-based Management-System muss aktiv sein!
Beachten Sie, dass als Voraussetzung für die IEC-61131-3-Programmierung des Controllers über ETHERNET das Kontrollkästchen **CoDeSys** im Web-based Management-System auf der Seite „Port“ aktiviert sein muss (Default-Einstellung).
Alternativ können Sie Client-PC und Controller zur Programmierung jedoch auch seriell mit einem Programmierkabel verbinden.

Die Beschreibung der Programmierung mit WAGO-I/O-*PRO* CAA ist nicht Bestandteil dieses Handbuchs. In den folgenden Kapiteln wird vielmehr auf wichtige Hinweise bei der Projekterstellung in der WAGO-I/O-*PRO* CAA und auf spezielle Bausteine hingewiesen, die Sie explizit für die Programmierung des Controllers nutzen können.

Ferner wird beschrieben, wie Sie in WAGO-I/O-*PRO* CAA einen geeigneten Kommunikationstreiber laden, ein IEC-61131-3-Programm auf den Controller übertragen und deren Abarbeitung starten.

Information



Weitere Information
Eine detaillierte Beschreibung der Software-Bedienung entnehmen Sie dem Handbuch „WAGO-I/O-*PRO* CAA“. Dieses finden Sie auf der Internetseite http://www.wago.com unter Dokumentation → WAGO-I/O-SYSTEM 759 → WAGO-I/O-*PRO* → 759-333

1. Starten Sie die Programmierumgebung unter **Startmenü \ Programme \ WAGO-I/O-PRO** und **WAGO-I/O-*PRO* CAA**.

2. Legen Sie unter **Datei / Neu** ein neues Projekt an.

Sie erhalten ein Dialogfenster, in dem Sie das Zielsystem für die Programmierung einstellen.

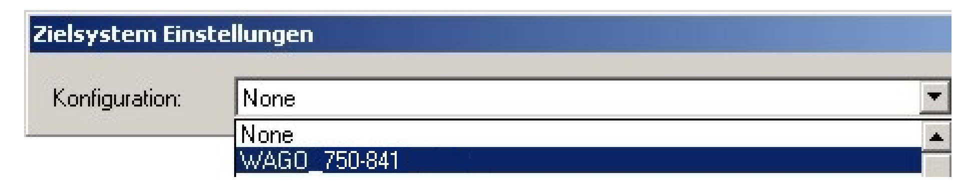


Abbildung 52: Dialogfenster Zielsystemeinstellungen

3. Wählen Sie den Feldbuscontroller 750-841 mit dem Eintrag **WAGO_750-841** aus und bestätigen Sie mit **OK**.



4. Wählen Sie im folgenden Dialogfenster die Programmierart (AWL, KOP, FUP, AS, ST oder CFC) aus.

Damit Sie in Ihrem neuen Projekt definiert auf alle Busklemmendaten zugreifen können, ist zunächst die Busklemmenkonfiguration gemäß der vorhandenen Feldbusknoten-Hardware zusammenzustellen und in einer Konfigurationsdatei „EA-config.xml“ abzubilden.

In dieser Datei wird festgelegt, ob der Schreibzugriff auf die Klemmen vom IEC-61131-3-Programm, von MODBUS/TCP oder von EtherNet/IP aus erfolgen darf.

Die Generierung der Datei kann, wie nachfolgend beschrieben, über die Konfiguration mit dem WAGO-I/O-Konfigurator erfolgen.

9.1 Feldbuscontroller mit dem I/O-Konfigurator konfigurieren

Der I/O-Konfigurator ist ein in der WAGO-I/O-*PRO* CAA eingebundenes PlugIn zum Ermitteln von Adressen für die Busklemmen an einem Controller.

1. Wählen Sie im linken Bildschirmfenster der WAGO-I/O-*PRO*-CAA-Oberfläche die Registerkarte **Ressourcen**.

2. Um den I/O-Konfigurator zu starten, klicken Sie in der Baumstruktur auf **Steuerungskonfiguration**.

3. Erweitern Sie in der Baumstruktur den Zweig **Hardware configuration** und anschließend den Unterzweig **K-Bus**.

4. Klicken Sie mit der rechten Maustaste auf den Eintrag **K-Bus**, so dass sich das Kontextmenü zum Einfügen und Anhängen weiterer Busklemmen öffnet.

5. Klicken Sie im Kontextmenü auf **Unterelement anhängen**, um den „I/O Module Catalogue“ zu öffnen. (In neueren Versionen des I/O-Konfigurators klicken Sie zusätzlich auf den Button **Hinzufügen** um den „I/O Module Catalogue“ zu öffnen.)

6. Wählen Sie die einzufügende Busklemme aus dem „I/O Module Catalogue“ aus und hängen Sie diese mittels **Insert** >> und **OK** an das Ende der Klemmenbus-Struktur an.

7. Um eine gewünschte Busklemme direkt vor einer anderen Busklemme in die Klemmenbusstruktur einzufügen, klicken Sie mit der rechten Maustaste auf eine Busklemme und anschließend auf **Element einfügen**. Der Befehl „Unterelement anhängen“ ist in diesem Fall deaktiviert.

Die entsprechenden Befehle erreichen Sie auch im Menü **Einfügen** in der Menüleiste des Hauptfensters. Sowohl **Unterelement anhängen** als auch **Element einfügen** öffnen den Dialog „I/O-Configuration“ zur Auswahl der Busklemmen. In diesem Dialog positionieren Sie alle gewünschten Klemmen in Ihre Knotenkonfiguration.

8. Positionieren Sie alle notwendigen Busklemmen in der korrekten Reihenfolge, bis diese mit der Konfiguration des physikalischen Knotens übereinstimmen.

Vervollständigen Sie auf diese Weise die Baumstruktur in der Hardware-Konfiguration. Berücksichtigen Sie alle Busklemmen, die Daten liefern oder erwarten.



Hinweis



Der Klemmenbusaufbau im WAGO-I/O-Konfigurator muss mit dem physikalischen Knotenaufbau übereinstimmen!
Die Anzahl der Busklemmen, die Daten liefern oder erwarten, muss unbedingt mit der vorhandenen Hardware übereinstimmen (ausgenommen sind z. B. Potentialeinspeise-, Vervielfältigungs- und Endklemmen). Die Anzahl der Ein-/Ausgangsbits oder -bytes der einzelnen angeschalteten Busklemmen entnehmen Sie den entsprechenden Beschreibungen der Busklemmen.

Information



Weitere Information
Um das Datenblatt einer Busklemme zu öffnen, klicken Sie im „I/O Module Catalogue“ (Dialog zum Anhängen der Busklemmen) auf die betreffende Busklemme und drücken die Schaltfläche **Data Sheet**. Das Datenblatt wird in einem neuen Fenster angezeigt.
Alle aktuellen Datenblätter finden Sie auf der Internetseite http://www.wago.com unter Service → Dokumentation.

9. Mit der Schaltfläche **OK** übernehmen Sie die Knotenkonfiguration und schließen den Dialog.

Die Adressen der Steuerungskonfiguration werden neu berechnet und die Baumstruktur der Steuerungskonfiguration aktualisiert.

Ändern Sie nun gegebenenfalls die gewünschte Zugriffsberechtigung für einzelne Busklemmen, wenn auf diese über einen Feldbus (z. B. MODBUS TCP/IP oder EtherNet/IP) zugegriffen werden soll. Zunächst ist für jede eingefügte Busklemme der Schreibzugriff vom PLC aus festgelegt. Um dieses zu ändern, führen Sie folgende Schritte aus:

10. Klicken Sie auf eine eingebundene Busklemme

11. Bestimmen Sie in dem rechten Dialogfenster unter Register „Modulparameter“ für jede einzelne Busklemme, von wo aus der Zugriff auf die Klemmendaten erfolgen soll.

Hierbei haben Sie in der Spalte „Wert“ folgende Auswahl:

- PLC (Standardeinstellung) - Zugriff vom PFC aus
- fieldbus 1 - Zugriff von MODBUS/TCP aus
- fieldbus 2 - Zugriff von Ethernet/IP aus

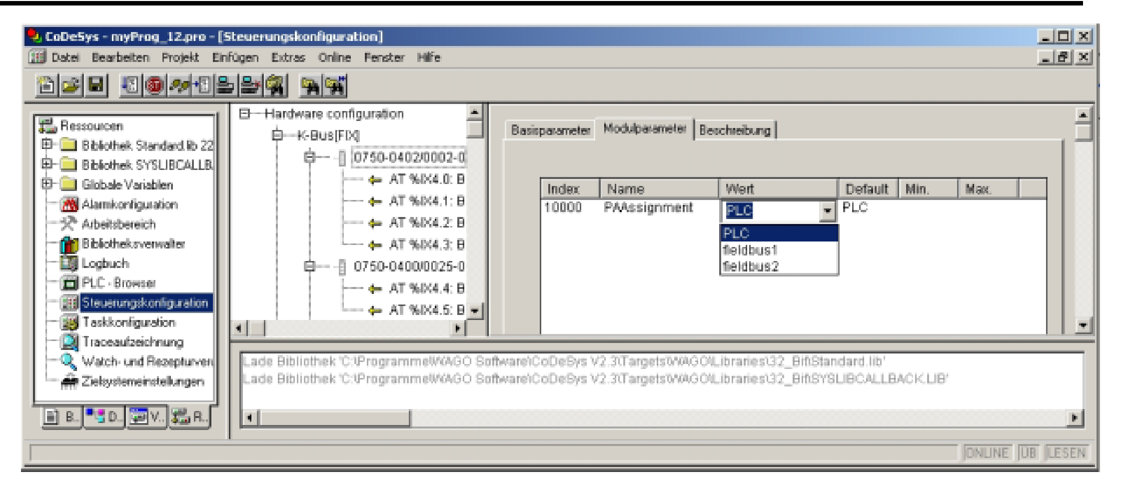


Abbildung 53: Schreibzugriff über Modulparameter

Nach Fertigstellung der Zuordnung, können Sie mit der IEC-61131-3-Programmierung beginnen.

Wenn Sie das Projekt übersetzen (Menü **Projekt > Übersetzen/Alles übersetzen**) und in den Feldbuscontroller laden, wird in dem Feldbuscontroller automatisch eine Konfigurationsdatei „EA-config.xml" generiert und abgelegt.

Hinweis



Bei direktem Schreiben über MODBUS an eine Hardware-Adresse „fieldbus1" einstellen!
Wenn Sie über MODBUS direkt auf eine Hardware-Adresse schreiben wollen, stellen Sie den Zugriff über „fieldbus1" ein. Ansonsten sind die Klemmen der SPS zugeordnet und es ist kein Schreiben von Außen möglich.

Information



Weitere Information
Eine detaillierte Beschreibung zur Bedienung der Software WAGO-I/O-*PRO* CAA und des I/O-Konfigurators finden Sie auch in der Online-Hilfe zur WAGO-I/O-*PRO* CAA.

9.1.1 Feldbuscontroller mit der Datei „EA-config.xml" konfigurieren

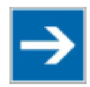
Sie können die Konfigurationsdatei „EA-config.xml" neben der automatischen Generierung in WAGO-I/O-*PRO* CAA auch manuell anlegen und ändern.

Die Datei legen Sie per FTP in dem Verzeichnis „/etc" auf dem Controller ab.

Im Folgenden wird die Konfiguration des Controllers mittels der Konfigurationsdatei „EA-config.xml" beschrieben.



Hinweis



Konfigurationseinträge in WAGO-I/O-*PRO* CAA überschreiben „EA-config.xml“ bei Download!
Wenn Sie die Klemmenzuordnung direkt mittels der im Feldbuscontroller gespeicherten Datei „EA-config.xml“ vornehmen, dürfen Sie zuvor keine Konfigurationseinträge in der WAGO-I/O-*PRO* CAA speichern, da die Datei durch die Einträge in der WAGO-I/O-*PRO* CAA bei jedem Download überschrieben wird.

1. Stellen Sie eine Verbindung via FTP zu Ihrem Controller her. Dazu verwenden Sie ein FTP-Programm oder geben in Ihrem Browser folgende Zeile ein:

   ftp://[IP-Adresse des Controller], z. B. ftp://192.168.1.201

2. Anschließend geben Sie den Benutzernamen **admin** sowie das Kennwort **wago** ein.

Die Datei „EA-config.xml“ liegt im Ordner „/etc“ auf dem Controller.

3. Kopieren Sie die Datei in ein lokales Verzeichnis auf Ihrem PC und öffnen Sie diese in einem beliebigen Editor (z. B. „WordPad“).

Die folgende Syntax ist bereits in der Datei vorbereitet:

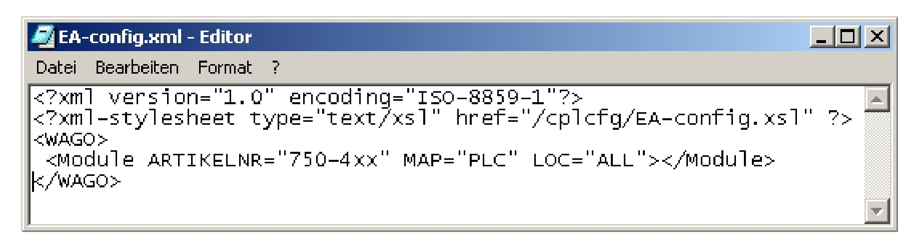
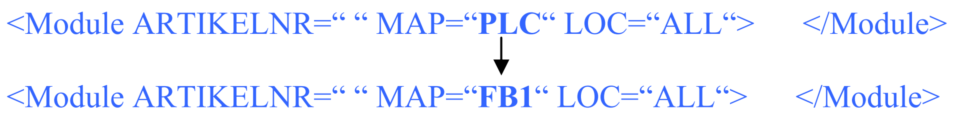


Abbildung 54: Konfigurationsdatei „EA-config.xml“

Die vierte Zeile enthält die notwendigen Informationen für die erste Busklemme. Der Eintrag MAP=„PLC“ weist dem IEC-61131-3-Programm die Schreibzugriffsrechte für das erste Modul zu.

4. Wenn Sie den Zugriff über MODBUS/TCP ermöglichen wollen, ersetzen Sie „PLC“ durch „FB1“ und für den Zugriff von Ethernet/IP aus durch „FB2“:



5. Um weitere Busklemmen hinzuzufügen, ergänzen Sie unter der vierten Zeile für jede einzelne Ihrer montierten Busklemmen eine neue Zeile in derselben Syntax und setzen Sie entsprechende Zugriffsberechtigungen.

Hinweis



Anzahl der Zeileneinträge gleich der Anzahl verwendeter Busklemmen!
Die Anzahl der Zeileneinträge muss unbedingt mit der Anzahl der vorhandenen Busklemmen in der Hardware-Konstellation übereinstimmen.

6. Speichern Sie die Datei und laden Sie diese wieder über den FTP-Client in das Dateisystem des Controllers.

Im Anschluss daran beginnen Sie mit der IEC-61131-3-Programmierung.

Information



Weitere Information
Eine detaillierte Beschreibung der Software-Bedienung entnehmen Sie dem Handbuch WAGO-I/O-*PRO* CAA. Dieses finden Sie auf der Internetseite http://www.wago.com unter Dokumentation → WAGO-I/O-SYSTEM 759 → WAGO-I/O-PRO → 759-333

9.2 ETHERNET-Bibliotheken für WAGO-I/O-*PRO* CAA

Für unterschiedliche IEC-61131-3-Programmieraufgaben stehen Ihnen in WAGO-I/O-*PRO* CAA verschiedene Bibliotheken zur Verfügung. Diese enthalten universell einsetzbare Funktionsbausteine und können somit Ihre Programmerstellung erleichtern und beschleunigen.

Information



Weitere Information
Sie finden alle Bibliotheken auf der Installations-CD zur Software WAGO-I/O-*PRO* CAA in dem Ordnerverzeichnis:
CoDeSys V2.3\Targets\WAGO\Libraries\...

Einige Bibliotheken z. B. „standard.lib“ und „IECsfc.lib“ sind standardmäßig eingebunden. Die in der nachfolgenden Liste beschriebenen Bibliotheken sind hingegen spezifisch für ETHERNET-Projekte. Sie finden die Bibliotheken auf der CD-ROM zu WAGO-I/O-PRO CAA.

Nach dem Einbinden der Bibliotheken stehen Ihnen enthaltene Funktionsbausteine, Funktionen und Datentypen zur Verfügung, die Sie genauso benutzen können, wie selbstdefinierte.

Tabelle 39: ETHERNET-Bibliotheken für WAGO-I/O-*PRO* CAA

Bibliothek	Beschreibung
Ethernet.lib	Funktionsbausteine zur Kommunikation via ETHERNET
WAGOLibEthernet_01.lib	Funktionsbausteine zur Herstellung einer Verbindung mit einem Remote-Server oder Client-PC (über TCP) und zum Datenaustausch mit jedem möglichen UDP-Server oder Client-PC (über UDP)
WAGOLibModbus_IP_01.lib	Funktionsbausteine zur Herstellung einer Verbindung mit einem oder mehreren Slaves
ModbusEthernet_04.lib	Funktionsbausteine für den Datenaustausch mit mehreren MODBUS/TCP/UDP-Slaves Stellt außerdem einen MODBUS-Server zur Verfügung, welcher die MODBUS-Dienste auf einem Word-Array abbildet.
SysLibSockets.lib	Funktionsbausteine für den Zugriff auf Sockets zur Kommunikation über TCP/IP und UDP
WagoLibSockets.lib	Funktionsbausteine für den Zugriff auf Sockets zur Kommunikation über TCP/IP und UDP Enthält im Gegensatz zu SysLibSockets.lib weitere Funktionen.
Mail_02.lib	Funktionsbausteine zum Versenden von Emails
WAGOLibMail_02.lib	Funktionsbausteine zum Versenden von Emails
WagoLibSnmpEx_01.lib	Funktionsbausteine zum Versenden von SNMP-V1-Traps zusammen mit Parametern des Typs DWORD und STRING(120) (ab Software-Version SW >= 07)
WagoLibSntp.lib	Funktionsbausteine zur Einstellung und Verwendung des Simple-Network-Time-Protocols (SNTP)
WagoLibFtp.lib	Funktionsbausteine zur Einstellung und Verwendung des File-Transfer-Protocols (FTP)
WAGOLibTerminalDiag.lib	Funktionsbausteine zur Ausgabe von Modul-, Kanal- und Diagnosedaten von Klemmen, die Diagnosedaten bereitstellen

Information



Weitere Information
Eine detaillierte Beschreibung der Bausteine und der Software-Bedienung entnehmen Sie dem Handbuch WAGO-I/O-*PRO* CAA auf der Internetseite http://www.wago.com unter Dokumentation → WAGO-I/O-SYSTEM 759 → WAGO-I/O-PRO → 759-333 oder der Online-Hilfe der WAGO-I/O-*PRO* CAA.

9.3 Einschränkungen im Funktionsumfang

Die Basis von WAGO-I/O-*PRO* CAA, das Standard Programmiersystem CoDeSys von 3S, besitzt eine integrierte Visualisierung. Diese Visualisierung kann je nach Zielsystem in den Varianten „HMI“, „TargetVisu“ und „WebVisu“ genutzt werden.

Der Feldbuscontroller unterstützt die Ablaufvarianten „HMI“ und „WebVisu“. Abhängig von der Ablaufvariante ergeben sich technologische Einschränkungen.

Verschiedene Optionen der komplexen Visualisierungsobjekte „Alarm“ und „Trend“ sind ausschließlich in der Einstellung „HMI“ verfügbar. Dieses gilt z. B. für das Versenden von Mails als Reaktion auf einen Alarm oder für die Navigation durch historische Trenddaten sowie deren Erzeugung.

Auf dem Feldbuscontroller wird die „WebVisu“ im Vergleich zur „HMI“ in wesentlich engeren physikalischen Grenzen ausgeführt. Kann die „HMI“ auf die nahezu unbeschränkten Ressourcen eines PCs zurückgreifen, muss die „WebVisu“ die folgenden Einschränkungen berücksichtigen:

Dateisystem (1,4 MB):
Die Gesamtgröße von SPS-Programm, Visualisierungsdateien, Bitmaps, Logdateien, Konfigurationsdateien usw. muss in das Dateisystem passen.
Die Größe des freien Speicherplatzes liefert der PLC-Browser auf das Kommando „fds“ (FreeDiscSpace)

Prozessdatenbuffer (16 kB):
Die WebVisu verwendet ein eigenes Protokoll für den Austausch von Prozessdaten zwischen Applet und Steuerung. Dabei werden die Prozessdaten ASCII-kodiert übertragen. Als Trennzeichen zwischen zwei Prozesswerten dient das Pipe-Zeichen ("|"). Damit ist der Platzbedarf einer Prozessdatenvariablen im Prozessdatenbuffer nicht nur abhängig vom Datentyp, sondern zusätzlich vom Prozesswert selbst. So belegt eine Variable vom Type "WORD" zwischen einem Byte für die Werte 0..9 und fünf Bytes für Werte ab 10000. Das gewählte Format erlaubt lediglich eine grobe Abschätzung des Platzbedarfes für die einzelnen Prozessdaten im Prozessdatenbuffer. Wird die Größe überschritten arbeitet die WebVisu nicht mehr erwartungsgemäß.

Bausteinanzahl (standardmäßig 512):
Die Gesamtgröße des SPS-Programmes wird unter anderem durch die maximale Bausteinanzahl bestimmt. Dieser Wert ist in den Zielsystemeinstellungen konfigurierbar.

Rechenleistung/Prozessorzeit:
Ein Programmierbarer Feldbuscontroller ETHERNET TCP/IP basiert auf einem Echtzeitbetriebssystem mit präemptivem Multitasking. Dabei verdrängen hochpriore Prozesse, wie z. B. das SPS-Programm, niederpriore Prozesse, wie z. B. Webserver.
Der Webserver liefert Prozessdaten und Applets für die Web Visualisierung.
Bei der Task-Konfiguration ist darauf zu achten, dass für alle Prozesse genügend

Prozessorzeit zur Verfügung steht. Die Task-Aufrufoption „freilaufend“ ist in Verbindung mit der „WebVisu“ nicht geeignet, da in diesem Fall das hochpriore SPS-Programm den Webserver verdrängt. Stattdessen sollte die Task-Aufrufoption „zyklisch“ mit einem realistischen Wert verwendet werden.
Einen Überblick über die realen Ausführungszeiten aller Tasks in WAGO-I/O-*PRO* CAA liefert der PLC-Browser auf das Kommando „tsk“.
Werden in einem SPS-Programm Betriebsystemfunktionen wie z. B. für das Handling von Sockets oder dem Dateisystem verwendet, werden diese Ausführungszeiten vom Kommando „tsk“ nicht berücksichtigt.

Zähler CTU:
Der Zähler CTU arbeitet im Wertebereich von 0 bis 32767.

Netzwerkbelastung:
Ein Programmierbarer Feldbuscontroller ETHERNET TCP/IP hat genau eine CPU, die sowohl für die Abarbeitung des SPS-Programms als auch für die Abwicklung des Netzwerkverkehrs zuständig ist. Die Kommunikation über ETHERNET verlangt, dass jedes empfangene Telegramm, unabhängig für wen es bestimmt ist, bearbeitet wird.

Eine deutliche Reduzierung der Netzwerkbelastung ist durch die Verwendung von „Switches“ statt „Hubs“ erreichbar.

Broadcast-Telegramme lassen sich allerdings nur beim Sender oder durch konfigurierbare Switches, die über Broadcast-Limiting verfügen, eindämmen. Ein Netzwerkmonitor wie z. B. Wireshark (www.wireshark.org) verschafft einen Überblick über die aktuelle Auslastung im Netzwerk.

Hinweis



Bandbreitenbegrenzung nicht zur Erhöhung der Betriebssicherheit verwenden!
Beachten Sie, dass die im Web-based Management-System unter dem Link „Ethernet“ konfigurierbare Bandbreitenbegrenzung kein geeignetes Mittel ist, um die Betriebssicherheit der „WebVisu“ zu erhöhen, da dabei Telegramme ignoriert bzw. verworfen werden.

Information



Weitere Information
Da die Definition von harten Eckdaten aus den oben genannten Gründen nicht möglich ist, nehmen Sie als Unterstützung für Ihre Planung die im Internet veröffentlichten Anwendungshinweise. Dort finden Sie entsprechende Projekte, welche die Leistungsfähigkeit der Webvisualisierung zeigen. Die Anwendungshinweise Sie auf der Internetseite http://www.wago.com unter Service → Downloads → AUTOMATION oder Gebäudetechnik



Hinweis



Maximale Schreibzyklen des EEPROM beachten!
Feldbuskoppler/-controller speichern einige Informationen wie IP-Adresse und IP-Parameter im EEPROM, damit diese nach einem Neustart verfügbar sind. Die Speicherzyklen eines EEPROM sind generell begrenzt. Oberhalb einer Grenze von etwa 1 Million Schreibzyklen kann der Speicher nicht mehr zugesichert werden. Ein defektes EEPROM macht sich erst bei einem Neustart durch Software-Reset oder Power-ON bemerkbar.
Der Feldbuskoppler/-controller startet dann aufgrund einer fehlerhaften Checksumme im EEPROM immer wieder mit den Default-Parametern.

Folgende Funktionen verwenden das EEPROM:

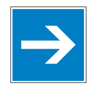
- **WAGO-I/O-PRO CAA**
  - **WagoLibDaylightSaving** SetDaylightSavings
  - **EthernetLib** SetNetworkConfig
SetVariables
- **MODBUS**
  - Register 0x1035 Time Offset
  - Register 0x100B Watchdog Parameter
  - Register 0x1028 Netzwerk Konfiguration
  - Register 0x1036 Daylight Saving
  - Register 0x1037 Modbus Response Delay
  - Register 0x2035 PI-Parameter
  - Register 0x2043 Default Konfiguration
- **Ethernet/IP**
  - Klasse 0xF5
  - Klasse 0xF6
  - Klasse 0x64
- **Parameterzuweisungen**
  - **BootP** Neue Parameter
  - **DHCP** Neue Parameter
  - **WAGO-MIB** Schreibzugriffe



9.4 Generelle Hinweise zu den IEC-Tasks

Beachten Sie bei der Programmierung Ihrer IEC-Tasks die folgenden Hinweise:

Hinweis



Unterschiedliche Prioritäten für IEC-Tasks verwenden!
Versehen Sie IEC-Tasks (in WAGO-I/O-*PRO* CAA unter Register **Ressourcen** > **Task-Konfiguration**) mit unterschiedlichen Prioritäten, ansonsten kommt es beim Übersetzen des Anwenderprogramms zu einer Fehlermeldung.

Unterbrechung von IEC-Tasks durch Tasks höherer Priorität möglich!
Eine laufende IEC-Task kann durch Tasks mit höherer Priorität in ihrer Ausführung unterbrochen werden. Erst wenn keine Task mit höherer Priorität mehr zur Ausführung ansteht, wird die Ausführung der unterbrochenen Task wieder aufgenommen

Verfälschung von Variablen in überlappenden Bereichen des Prozessabbildes!
Verwenden mehrere IEC-Tasks Ein-/Ausgangsvariablen mit gleichen oder überlappenden Adressen im Prozessabbild, können sich die Werte der Ein-/Ausgangsvariablen während der Ausführung der IEC-Task ändern!

Wartezeit bei freilaufenden Tasks beachten!
Für Controller mit der Firmwareversion FW ≤ 10:
Freilaufende Tasks werden nach jedem Task-Zyklus für 1 ms angehalten, bevor die erneute Ausführung beginnt.

Für Controller mit der Firmwareversion FW >10:
Freilaufende Tasks werden nach jedem Task-Zyklus für die Hälfte der Zeit angehalten, welche die jeweilige Task selbst benötigt (mind. 1 ms).
Danach beginnt die erneute Ausführung.

Beispiel: 1. Task 4 ms → Wartezeit 2 ms
2. Task 2 ms → Wartezeit 1 ms

Default-Task wird standardmäßig angelegt!
Auch wenn Sie in WAGO-I/O-*PRO* CAA im Register **Ressourcen > Task-Konfiguration** keine Task anlegen, wird automatisch eine freilaufende „DefaultTask" erstellt.
Achten Sie darauf, für Ihre Task nicht denselben Namen zu verwenden.



Watchdog-Empfindlichkeit für zyklische Tasks beachten!
Die Watchdog-Empfindlichkeit gibt an, bei wie vielen Überschreitungen der Watchdog-Zeit ein Ereignis ausgelöst wird. Die Empfindlichkeit stellen Sie in WAGO-I/O-*PRO* CAA unter Register **Ressourcen > Task-Konfiguration** für zyklische Tasks ein. Eine Empfindlichkeit von 0 oder 1 ist gleichbedeutend und bewirkt, dass bei einmaliger Überschreitung der Watchdog-Zeit das Watchdog-Ereignis ausgelöst wird.
Bei einer Empfindlichkeit von 2, wird beispielsweise ein Watchdog-Ereignis ausgelöst, wenn in zwei aufeinanderfolgenden Taskzyklen die Watchdog-Zeit überschritten wird.

Für zyklische Tasks mit aktiviertem Watchdog zur Zykluszeitüberwachung gilt:

Hinweis



Hinweise zur Einstellung des Watchdogs!
Für jede angelegte Task kann ein Watchdog aktiviert werden, der die Abarbeitungszeit der jeweiligen Task überwacht.
Überschreitet die Tasklaufzeit die angegebene Watchdog-Zeit (z. B. t#200 ms), dann ist der Watchdog-Fall eingetreten.
Das Laufzeitsystem stoppt das IEC-Programm und meldet einen Fehler.

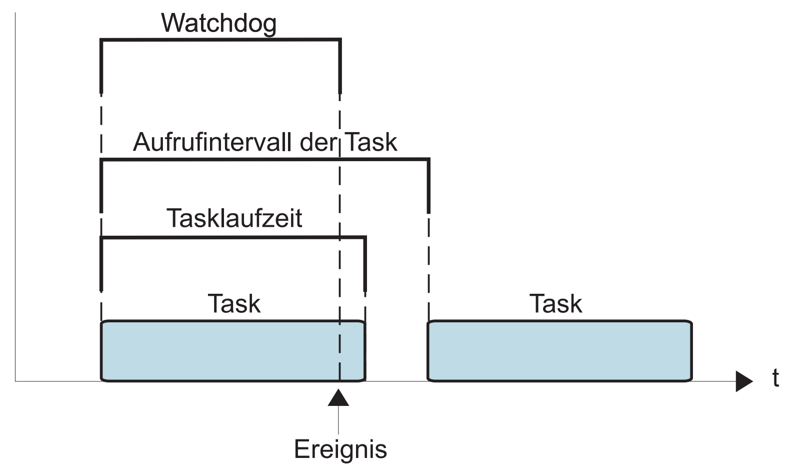


Abbildung 55: Watchdog-Laufzeit kleiner als Tasklaufzeit

Ist die eingestellte Watchdog-Zeit größer als das Aufrufintervall der Task, wird zu jedem Aufrufintervall der Watchdog neu gestartet.

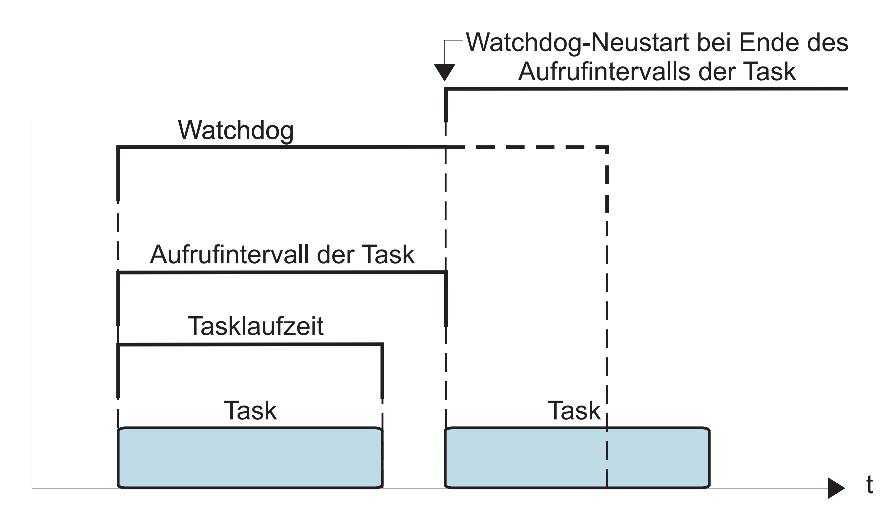


Abbildung 56: Watchdog-Laufzeit größer als Task-Aufrufintervall

Empfehlung:
Stellen Sie die Watchdog-Zeit größer als die Tasklaufzeit und kleiner als das Taskaufrufintervall ein.

Für zyklische Tasks gilt:



Hinweis



Keine zyklischen Tasks mit Aufrufintervall > 30 min. möglich!
Es sind keine zyklischen Tasks mit einem Aufrufintervall größer als 30 Minuten möglich.

9.4.1 Ablaufschema einer IEC-Task

1. Systemzeit ermitteln (tStart)
2. Wenn seit dem letzten Schreiben der Ausgänge noch kein vollständiger Klemmenbuszyklus gefahren wurde
   → Auf das Ende des nächsten Klemmenbuszyklus warten
3. Eingänge lesen und Ausgänge aus dem Prozessabbild zurücklesen
4. Wenn das Anwenderprogramm gestartet wurde
   → Programmcodes dieser Task ausführen
5. Ausgänge in das Prozessabbild schreiben
6. Systemzeit ermitteln (tEnd)
   → tEnd - tStart = Laufzeit der IEC-Task

9.4.2 Die wichtigsten Task-Prioritäten im Überblick

Tabelle 40: Task-Abarbeitung

Task	Wichtigkeit der Abarbeitung
Klemmenbus-Task, Feldbus-Task	vorrangig vor allen anderen
Normale Task	nach den Klemmenbus- und Feldbus-Tasks
PLC-Comm-Task	nach den Normalen Tasks
Background-Task	nach den PLC-Comm-Tasks

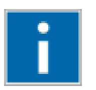
Klemmenbus-Task/Feldbus-Task (Intern)
Bei der Klemmenbus-Task handelt es sich um eine interne Task, die zyklisch das Prozessabbild mit den Ein-/Ausgangsdaten der Klemmen abgleicht.
Die Feldbus-Tasks laufen ereignisgesteuert und nehmen lediglich Rechenzeit in Anspruch, wenn über den Feldbus kommuniziert wird (MODBUS/Ethernet/IP).

Normale Task (in CoDeSys einstellbare IEC-Task Prioritäten 1-10)
IEC-Tasks mit dieser Priorität können durch die Klemmenbus-Task unterbrochen werden. Deshalb muss die gesteckte Klemmenkonfiguration und die Kommunikation über den Feldbus bei aktiviertem Watchdog für das Task-Aufrufintervall berücksichtigt werden.

PLC-Comm-Task (Intern)
Die PLC-Comm-Task ist im eingeloggten Zustand aktiv und übernimmt die Kommunikation mit dem Gateway der WAGO-I/O-*PRO* CAA.

Background-Task (in CoDeSys einstellbare IEC-Task Prioritäten 11-31)
Alle internen Tasks haben eine höhere Priorität als IEC-Background-Tasks. Von daher eignen sich diese Tasks besonders, um zeitintensive und zeitunkritische Aufgaben durchzuführen, beispielsweise zum Aufruf der Funktionen in der SysLibFile.lib.

Information



Weitere Information
Eine detaillierte Beschreibung zu dem Programmiertool WAGO-I/O-*PRO* CAA entnehmen Sie dem Handbuch WAGO-I/O-*PRO* CAA auf der Internetseite http://www.wago.com unter Dokumentation → WAGO-I/O-SYSTEM 759 → WAGO-I/O-PRO → 759-333



9.5 Systemereignisse

Anstelle einer Task kann auch ein Systemereignis (Event) einen Projektbaustein zur Abarbeitung aufrufen.

Die dazu verwendbaren Systemereignisse sind zielsystemabhängig. Sie setzen sich zusammen aus der Liste der unterstützten Standardsystemereignisse der Steuerung und eventuell hinzugefügten herstellerspezifischen Ereignissen.

Mögliche Ereignisse sind z. B. „Stop", „Start", „Online Change".

Die vollständige Liste aller Systemereignisse wird in WAGO-I/O-*PRO* CAA aufgeführt.

9.5.1 Systemereignisse aktivieren/deaktivieren

1. Öffnen Sie in WAGO-I/O-*PRO* CAA das Register **Ressourcen** > **Task-Konfiguration** > **Systemereignisse** (siehe folgende Abbildung).

2. Damit ein Baustein durch ein Ereignis aufgerufen werden kann, aktivieren Sie die gewünschten Einträge durch Setzen von Haken in die betreffenden Kontrollkästchen.

3. Deaktivieren Sie Kontrollkästchen, indem Sie die Haken mit einem Mausklick entfernen.

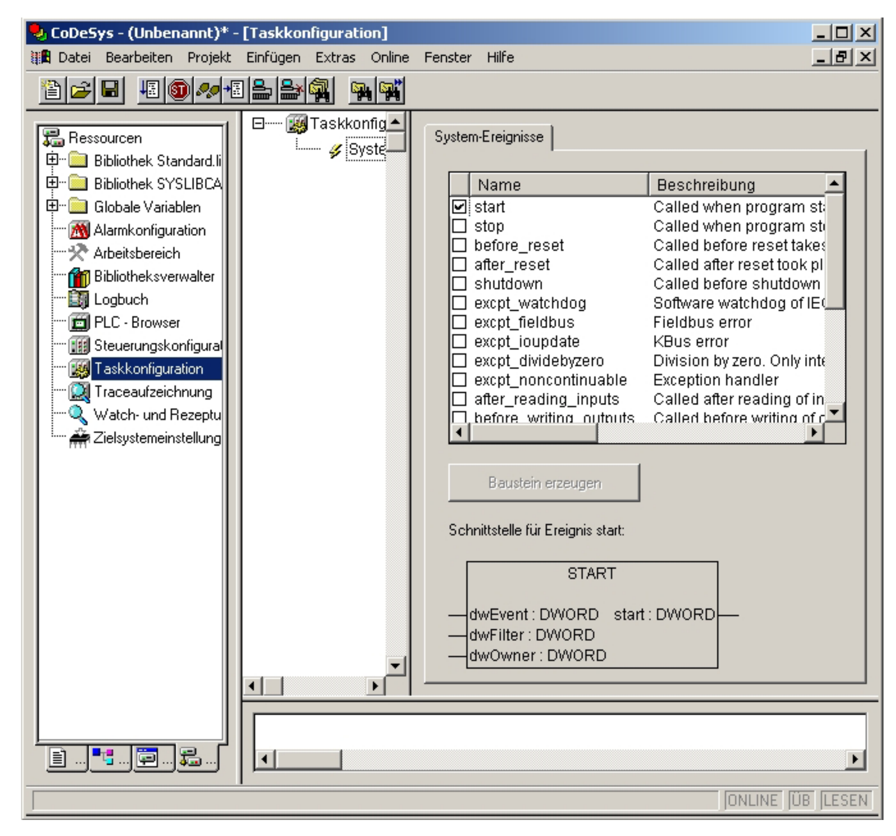


Abbildung 57: Systemereignisse aktivieren/deaktivieren

Information



Weitere Information
Die Zuordnung der Systemereignisse zu dem jeweils aufzurufenden Funktionsbaustein finden Sie detailliert beschrieben in dem Handbuch zum Programmiertool WAGO-I/O-*PRO* CAA auf der Internetseite http://www.wago.com unter Dokumentation → WAGO-I/O-SYSTEM 759 → WAGO-I/O-PRO → 759-333

9.6 IEC-Programm auf den Controller übertragen

Sie können eine erstellte IEC-61131-Applikation auf zwei Arten von Ihrem PC auf den Controller übertragen (siehe folgende Kapitel):

- mittels serieller RS-232-Schnittstelle direkt übertragen
- mittels TCP/IP über den Feldbus übertragen

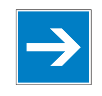
Für die Übertragung sind geeignete Kommunikationstreiber erforderlich, welche Sie in WAGO-I/O-*PRO* CAA laden und konfigurieren.

Hinweis



Kommunikationsparameter des Treibers anpassen!
Achten Sie bei der Auswahl des gewünschten Treibers auf die richtigen Einstellungen und Anpassungen der Kommunikationsparameter (siehe nachfolgende Beschreibung).

Hinweis



„Reset“ und „Start“ zum Setzen der physikalischen Ausgänge notwendig!
Die Initialisierungswerte für die physikalischen Ausgänge werden nicht direkt nach dem Download gesetzt. Wählen sie in der Menüleiste der WAGO-I/O-*PRO* CAA **Online** > **Reset** und nachfolgend **Online** > **Start** zum Setzen der Werte.

Hinweis



Applikation vor dem Erzeugen großer Bootprojekte stoppen!
Stoppen Sie vor dem Erzeugen eines sehr großen Bootprojektes die WAGO-I/O-PRO-CAA-Applikation mittels **Online** > **Stop**, da es sonst zu einem Stoppen des Klemmenbusses kommen kann. Nach dem Erzeugen des Bootprojektes können Sie die Applikation wieder starten.

Hinweis



Handling persistenter Daten beeinflusst den Programmstart!
In Abhängigkeit von Variablentyp, Anzahl und Größe der persistenten Daten sowie deren Kombination, z. B. in Funktionsbausteinen, kann das Handling mit persistenten Daten den Programmstart durch eine verlängerte Initialisierungsphase verzögern.

Information



Weitere Informationen
Die folgende Beschreibung dient dem schnellen Einstieg. Die Installation fehlender Kommunikationstreiber sowie die detaillierte Software-Bedienung entnehmen Sie dem Handbuch WAGO-I/O-*PRO* CAA auf der Internetseite http://www.wago.com → Service → Downloads → Dokumentation → WAGO Software 759 → WAGO-I/O-*PRO*/CoDeSys (Programmierung)

9.6.1 Applikation mittels serieller Schnittstelle übertragen

Hinweis



Stellung des Betriebsartenschalters bei Zugriff auf Controller beachten!
Für den Zugriff auf den Feldbuscontroller muss der Betriebsartenschalter, der sich hinter der Abdeck-Klappe des Feldbuscontrollers neben der Service-Schnittestelle befindet, in der mittleren oder in der oberen Stellung sein.

Um eine physikalische Verbindung über die serielle Service-Schnittstelle herzustellen, verwenden Sie das WAGO-Kommunikationskabel. Dieses ist im Lieferumfang der Programmiersoftware WAGO-I/O-*PRO* CAA (Art.-Nr.: 759-333) enthalten oder kann als Zubehör über die Bestell-Nr.: 750-920 bezogen werden.

ACHTUNG



Kommunikationskabel 750-920 nicht unter Spannung stecken!
Um Schäden an der Kommunikationsschnittstelle zu vermeiden, stecken und ziehen Sie das Kommunikationskabel 750-920 nicht unter Spannung!
Der Feldbuskoppler muss dazu spannungsfrei sein!

1. Kontrollieren Sie, ob sich der Betriebsartenschalter in der mittleren oder in der oberen Stellung befindet.
   Sollte dieses nicht der Fall sein, bringen Sie den Betriebsartenschalter in die mittlere oder obere Stellung.

2. Verbinden Sie über das WAGO-Kommunikationskabel eine COM-Schnittstelle Ihres PCs mit der seriellen Service-Schnittstelle des Feldbuscontrollers.

Für die serielle Datenübertragung ist ein Kommunikationstreiber erforderlich. Dieser Treiber und seine Parametrierung wird in WAGO-I/O-*PRO* CAA in dem Dialog „Kommunikationsparameter“ eingetragen:

3. Starten Sie die Software WAGO-I/O-*PRO* CAA unter **Startmenü** > **Programme** > **WAGO-Software** > **CoDeSys for Automation Alliance** > **CoDeSys V2.3** (oder andere Version).

4. Wählen Sie im Menü **Online** den Unterpunkt **Kommunikationsparameter** aus.

Der Dialog „Kommunikationsparameter“ öffnet sich. Auf der linken Seite des Dialoges werden die Kanäle der aktuell verbundenen Gateway-Server und darunter die bereits installierten Kommunikationstreiber angezeigt.
In der Grundeinstellung sind in diesem Dialog noch keine Einträge vorhanden.

5. Klicken Sie auf **Neu...**, um eine neue Verbindung herzustellen und vergeben Sie einen Namen, z. B. RS-232-Verbindung.



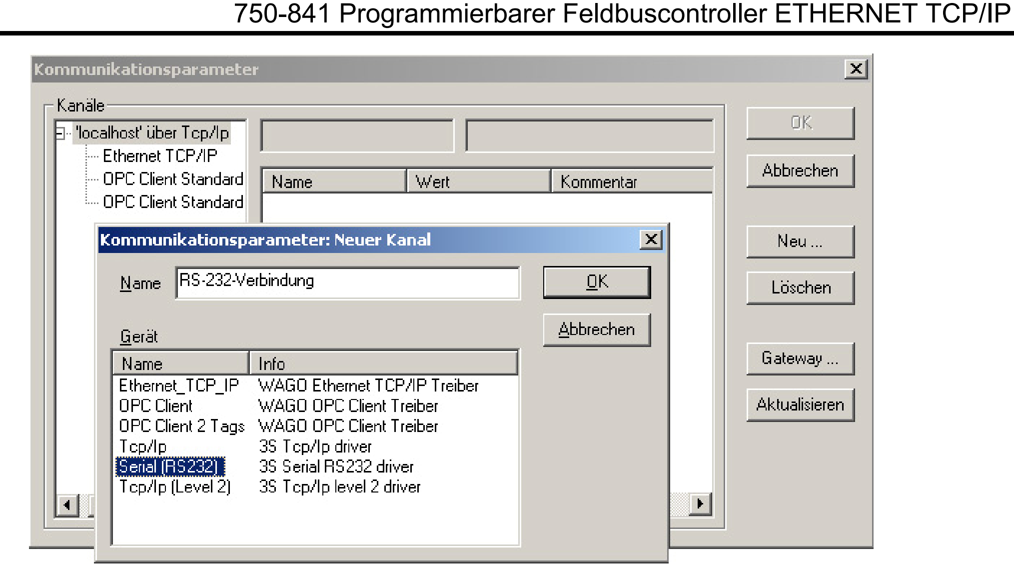


Abbildung 58: Dialogfenster „Kommunikationsparameter“, Erstellen einer neuen Verbindung

6. Markieren Sie in dem Auswahlfenster auf der rechten Seite des Dialogs den gewünschten Treiber **Serial (RS-232) – 3S Serial RS-232 driver**, um die serielle Verbindung zwischen PC und Feldbuscontroller zu konfigurieren.

In dem mittleren Fenster des Dialogs sind die folgenden Standardeinträge vorhanden:

- Port: COM1
- Baudrate: 19200
- Parity: Even
- Stop-bits: 1
- Motorola byteorder: No

7. Ändern Sie gegebenenfalls die Einträge entsprechend der obigen Werte ab, indem Sie auf den jeweiligen Wert klicken und diesen editieren.

8. Bestätigen Sie mit **OK**

Die RS-232-Schnittstelle ist nun für das Übertragen der Applikation konfiguriert.

9. Um eine Verbindung mit dem Feldbuscontroller aufzubauen, klicken Sie im Menü **Online** auf **Einloggen**.

Durch das Einloggen wird der Online-Modus zum Feldbuscontroller eingeschaltet und die Kommunikationsparameter sind nicht mehr aufrufbar.

Sofern noch kein Programm im Feldbuscontroller vorhanden ist, erscheint ein Fenster mit der Abfrage, ob das Programm geladen werden soll.

10. Um das aktuelle Programm zu laden, bestätigen Sie mit **Ja**.

11. Klicken Sie im Menü **Online** auf **Bootprojekt erzeugen**.

Auf diese Weise wird Ihr kompiliertes Projekt auch ausgeführt, wenn Sie den Feldbuscontroller neu starten oder wenn es einen Spannungsausfall gibt.

12. Wenn das Programm geladen ist, starten Sie die Programmabarbeitung über das Menü **Online** und den Menüpunkt **Start**.

Dieser Befehl startet die Abarbeitung Ihres Programms in der Steuerung bzw. in der Simulation.

Am rechten Ende der Statusleiste wird „ONLINE“ und „LÄUFT“ angezeigt.

13. Um den Online-Betrieb zu beenden, klicken Sie im Menü **Online** auf den Menüpunkt **Ausloggen**.



9.6.2 Applikation via ETHERNET übertragen

Die physikalische Verbindung zwischen PC und Feldbuscontroller erfolgt über das Feldbuskabel.
Für die Datenübertragung ist ein geeigneter Kommunikationstreiber erforderlich. Den Treiber und seine Parameter tragen Sie in WAGO-I/O-*PRO* CAA im Dialog „Kommunikationsparameter“ ein:

Hinweis



Feldbuscontroller benötigt IP-Adresse für den Zugriff!
Damit Sie auf den Feldbuscontroller zugreifen können, benötigt der Feldbuscontroller eine IP-Adresse. Der Betriebsartenschalter, der sich hinter der Abdeck-Klappe des Feldbuscontrollers neben der Service-Schnittestelle befindet, muss in der mittleren oder in der oberen Stellung sein.

1. Starten Sie die Software WAGO-I/O-*PRO* CAA unter **Startmenü** > **Programme** > **WAGO-Software** > **CoDeSys for Automation Alliance** > **CoDeSys V2.3** (oder andere Version).

2. Wählen Sie im Menü **Online** den Unterpunkt **Kommunikationsparameter** aus.

Der Dialog „Kommunikationsparameter“ öffnet sich. Auf der linken Seite des Dialoges werden die Kanäle der aktuell verbundenen Gateway-Server und darunter die bereits installierten Kommunikationstreiber angezeigt.
In der Grundeinstellung sind in diesem Dialog noch keine Einträge vorhanden.

3. Klicken Sie auf **Neu...**, um eine neue Verbindung herzustellen und vergeben Sie einen Namen, z. B. TcpIp-Verbindung.

4. Markieren Sie in dem Auswahlfenster auf der rechten Seite des Dialogs den gewünschten TCP/IP-Treiber, um die Verbindung zwischen PC und Feldbuscontroller via ETHERNET zu konfigurieren.
Verwenden Sie die neue Treiber-Version „**Tcp/Ip**“ (3S Tcp/Ip driver)**..**
(Wenn Sie zwischen TCP und UDP wählen möchten, verwenden Sie den Treiber „Ethernet_TCP_IP“ (WAGO-ETHERNET-TCP/IP- Treiber).

In dem mittleren Fenster des Dialogs sind die folgenden Standardeinträge vorhanden:

- Adresse: IP-Adresse des Feldbuscontrollers
- Port: 2455
- Motorolabyteorder: No

5. Ändern Sie gegebenenfalls die Einträge entsprechend der obigen Werte ab, indem Sie auf den jeweiligen Wert klicken und diesen editieren.

6. Bestätigen Sie mit **OK**.

Die TCP/IP-Schnittstelle ist nun für das Übertragen der Applikation konfiguriert.

7. Um eine Verbindung mit dem Feldbuscontroller aufzubauen, klicken Sie im Menü **Online** auf **Einloggen**.

Durch das Einloggen wird der Online-Modus zum Feldbuscontroller eingeschaltet und die Kommunikationsparameter sind nicht mehr aufrufbar.

Sofern noch kein Programm im Feldbuscontroller vorhanden ist, erscheint ein Fenster mit der Abfrage, ob das Programm geladen werden soll.

8. Um das aktuelle Programm zu laden, bestätigen Sie mit **Ja**.

9. Klicken Sie im Menü **Online** auf **Bootprojekt erzeugen**.

Auf diese Weise wird Ihr kompiliertes Projekt auch ausgeführt, wenn Sie den Feldbuscontroller neu starten oder wenn es einen Spannungsausfall gibt.

10. Wenn das Programm geladen ist, starten Sie die Programmabarbeitung über das Menü **Online** und den Menüpunkt **Start**.

Dieser Befehl startet die Abarbeitung Ihres Programms in der Steuerung bzw. in der Simulation.

Am rechten Ende der Statusleiste wird „ONLINE“ und „LÄUFT“ angezeigt.

11. Um den Online-Betrieb zu beenden, klicken Sie im Menü **Online** auf den Menüpunkt **Ausloggen**.



10 Im Web-based Management-System (WBM) konfigurieren

Für die Konfiguration und Verwaltung des Systems stehen Ihnen ein internes Dateisystem und ein integrierter Webserver zur Verfügung, die als Web-based Management-System, kurz WBM, bezeichnet werden.

Auf den intern gespeicherten HTML-Seiten erhalten Sie auslesbare Informationen über die Konfiguration und den Status des Feldbusknotens. Außerdem ändern Sie hier die Konfiguration des Gerätes.
Darüber hinaus können Sie über das implementierte Dateisystem auch selbst erstellte HTML-Seiten hinterlegen.

Hinweis



Nach Änderungen an der Konfiguration immer einen Neustart durchführen!
Damit geänderte Konfigurationseinstellungen wirksam werden, führen Sie nach Ihren Änderungen immer einen Systemneustart durch.

1. Zum Öffnen des WBM starten Sie einen Web-Browser (z. B. Microsoft Internet-Explorer oder Mozilla Firefox).

2. Geben Sie in der Adresszeile die IP-Adresse des Feldbuskopplers/-controllers ein (standardmäßig 192.168.1.1 oder wie zuvor konfiguriert).

3. Bestätigen Sie mit **[Enter]**.
   Die Startseite des WBM wird aufgebaut.

4. Wählen Sie in der linken Navigationsleiste den Link auf die gewünschte HTML-Seite.
   Es erscheint ein Abfragedialog.

5. Geben Sie im Abfragedialog Ihren Benutzernamen und das Passwort ein (standardmäßig: User = „admin“, Passwort = „wago“ oder User = „user“, Passwort = „user“).
   Die entsprechende HTML-Seite wird aufgebaut.

6. Führen Sie die gewünschten Einstellungen durch.

7. Bestätigen Sie Ihre Änderungen mit der Schaltfläche **[SUBMIT]** oder verwerfen Sie diese mit der Schaltfläche **[UNDO]**.

8. Damit die Einstellungen übernommen werden, führen Sie anschließend einen Neustart durch.

10.1 Information

Auf der Default-Startseite des WBM „Information“ erhalten Sie eine Übersicht mit allen wichtigen Informationen zu Ihrem Feldbuskoppler/-controller.

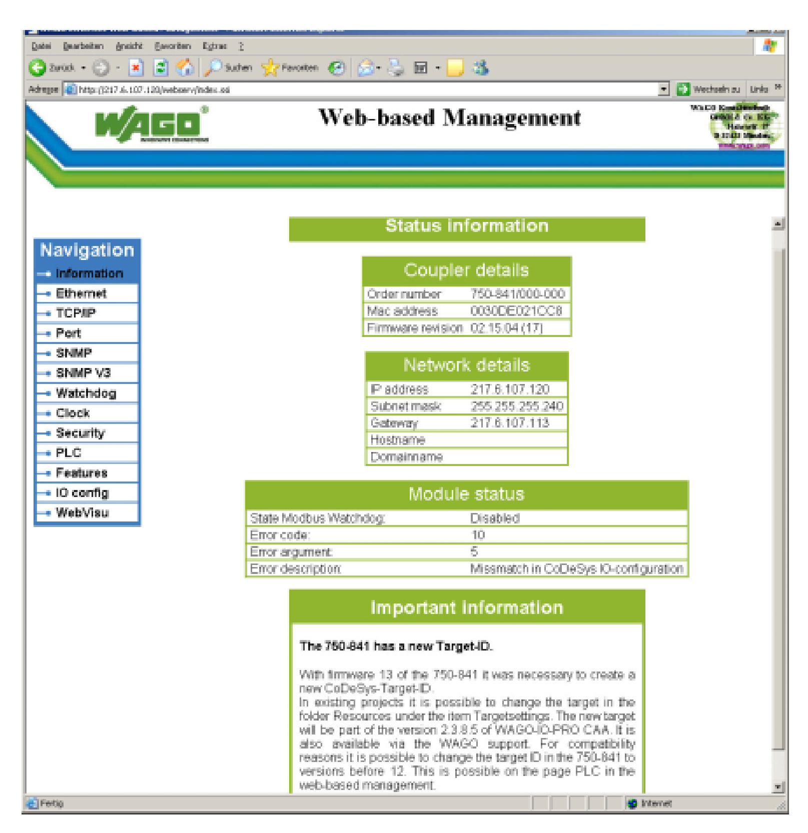
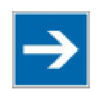


Abbildung 59: WBM-Seite „Information“

Tabelle 41: WBM-Seite „Information“

Coupler details			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
Order number	750-841/000-000	750-841/000-000	Artikelnummer
Mac address	0030DEXXXXXX	0030DE021CC8	Hardware MAC-Adresse
Firmware revision	kk.ff.bb (rr)	02.15.04 (17)	Firmware-Revisionsnummer (kk = Kompatibilität, ff = Funktionalität, bb = Bugfix, rr = Revision)
Network details			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
IP address	192.168.1.1	217.6.107.120	IP-Adresse,
Subnet mask	255.255.255.0	255.255.255.240	Subnetzmaske
Gateway	0.0.0.0	217.6.107.113	Gateway
Hostname	___	___	Hostname (hier nicht vergeben)
Domainname	___	___	Domainname (hier nicht vergeben)
Module status			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
State Modbus Watchdog	Disabled	Disabled	Status des Modbus Watchdog
Error code	0	10	Fehlercode
Error argument	0	5	Fehlerargument
Error description	Coupler running, OK	Missmatch in CoDeSysy IO-configuration	Fehlerbeschreibung

Hinweis



Für Kompatibilität bei Firmware-Version > FW11 Target-ID beachten!
Achten Sie bei Feldbuscontrollern mit einer Firmwareversion > FW 11 darauf, dass in Ihrem CoDeSys-Projekt die entsprechende Target-ID "750-841 (FW12-...)" ausgewählt ist. Bei Übernahme bereits bestehender älterer CoDeSys-Projekte, passen Sie gegebenenfalls die Target-ID in CoDeSys in dem Register "Ressourcen"/"Targetsettings" an, und kompilieren Sie das Projekt neu, um die Kompatibilität zu gewährleisten. Alternativ können Sie auf der WBM-Seite "PLC" die Funktion **Target ID – Use alternative Target ID** mit einem Häkchen in der Checkbox aktivieren.

10.2 Ethernet

Auf der Seite „Ethernet“ konfigurieren Sie die Übertragungsrate und die Bandbreitenbegrenzung für die ETHERNET-Kommunikation.

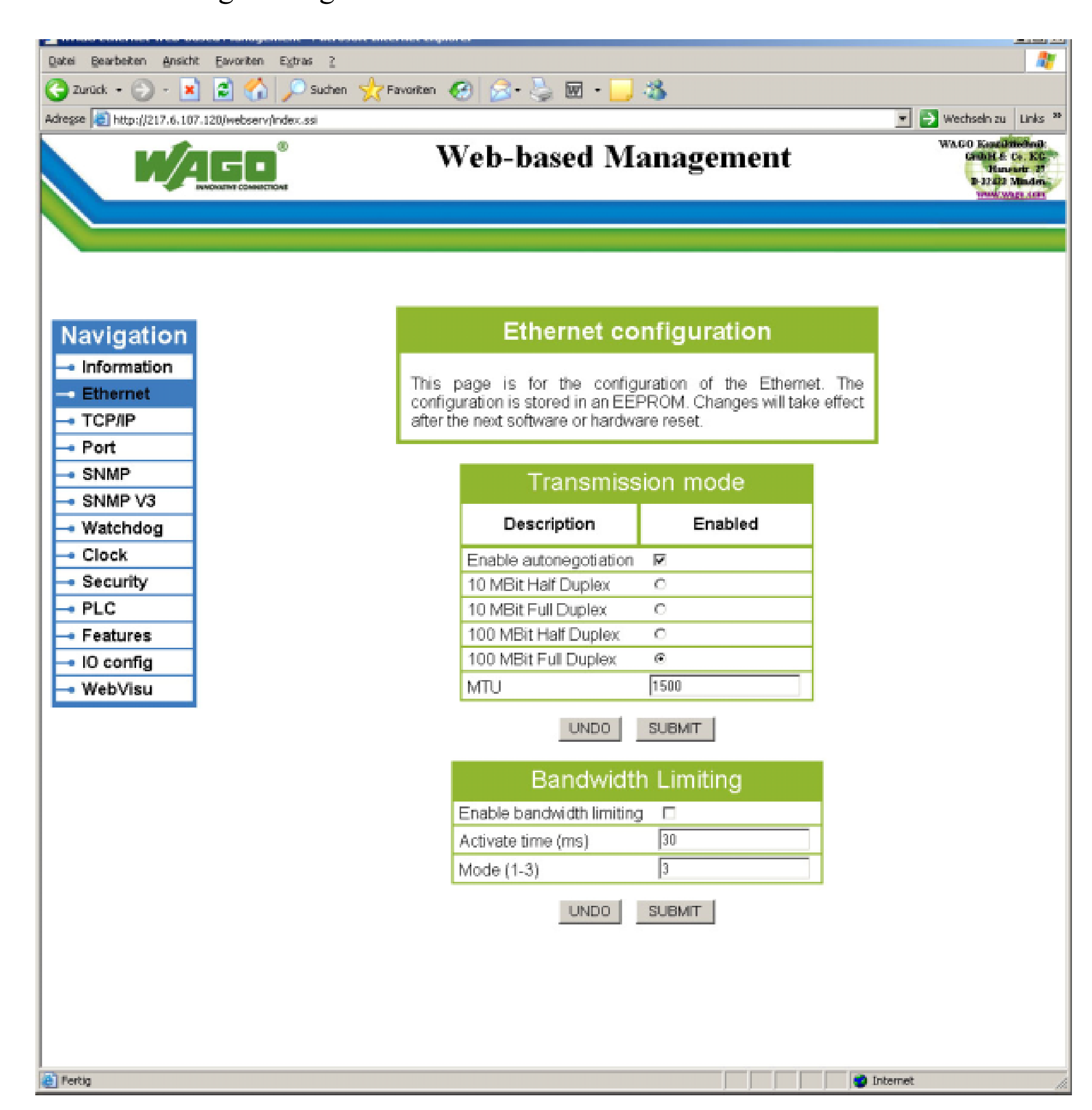


Abbildung 60: WBM-Seite „Ethernet“

Tabelle 42: WBM-Seite „Ethernet“

Transmission Mode		
Eintrag	**Standardwert**	**Beschreibung**
Enable autonegotiation	☑	☑ Autonegotiation aktivieren Der ETHERNET_Übertragungsmodus wird automatisch an den Kommunikationspartner angepasst. ☐ Autonegotiation deaktivieren
10 MBit Half Duplex 10 MBit Full Duplex 100 MBit Half Duplex 100 MBit Full Duplex	○ ○ ○ ⊙	Verwenden eines festen ETHERNET-Übertragungsmodus 10/100 MBit Halb-/ Vollduplex.
MTU	1500	Maximum Transmission Unit, maximale Paketgröße eines Protokolls, welche ohne Fragmentierung übertragen werden kann.
Bandwidth Limiting		
		Beschreibung
Enable bandwidth limiting	☐	☑ Bandbreitenbegrenzung aktivieren ☐ Bandbreitenbegrenzung deaktivieren
Activate time (ms)	30	Watchdog-Zeit, nach der die Interrupts gesperrt werden. Dabei können Telegramme verloren gehen. Der Timer wird von einer Task im System zyklisch neu gestartet.
Mode (1-3)	3	Prioritätsbestimmung der Task: Mode 1: Nur der Klemmenbus ist aktiv. Mode 2: Der Klemmenbus und die Feldbuscontroller-CPU bleiben aktiv. Mode 3: Alle Tasks werden aktiviert.

Hinweis



MTU-Wert nur für Fragmentierung einstellen!
Stellen Sie nur dann den Wert für MTU, d. h. die maximale, zwischen Client und Server vereinbarte Paketgröße, entsprechend ein, wenn Sie ein Tunnel-Protokoll (z. B. VPN) für die ETHERNET-Kommunikation verwenden und die Pakete fragmentiert werden müssen.
Diese Wert-Einstellung ist unabhängig von dem gewählten Übertragungsmodus.



Hinweis



ETHERNET-Übertragungsmodus korrekt konfigurieren!
Eine fehlerhafte Konfiguration des ETHERNET-Übertragungsmodus kann einen Verbindungsverlust, eine schlechte Netzwerk-Performance oder ein fehlerhaftes Verhalten des Feldbuskopplers/-controllers zur Folge haben.

10.3 TCP/IP

Auf der HTML-Seite „TCP/IP“ konfigurieren Sie die Netzwerkadressierung und die Netzwerkidentifikation.

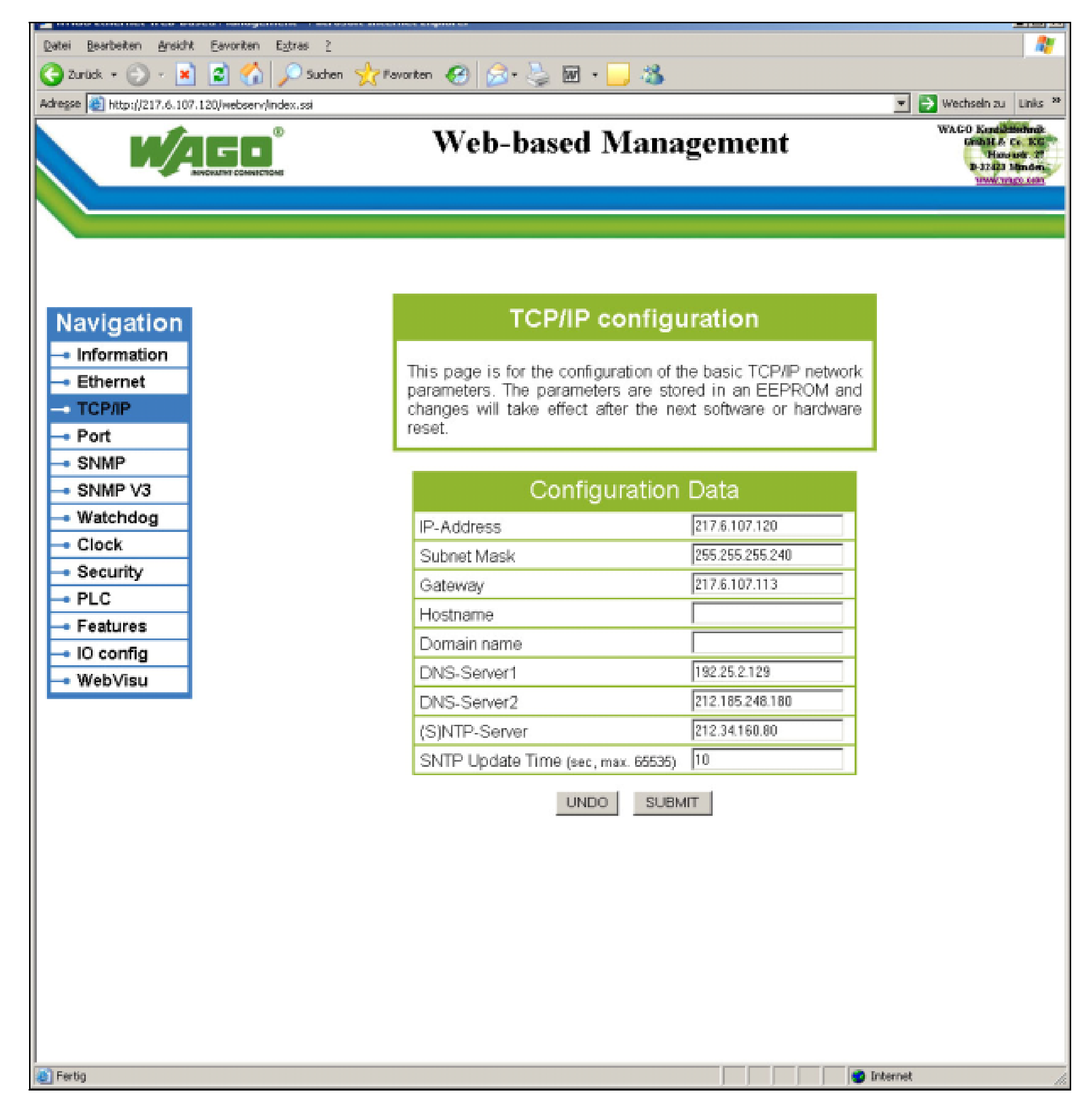


Abbildung 61: WBM-Seite „TCP/IP“

Tabelle 43: WBM-Seite „TCP/IP“

Configuration Data			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
IP-Address	192.168.1.0	192.168.1.200	IP-Adresse eintragen
Subnet mask	255.255.255.0	255.255.255.0	Subnetzmaske eintragen
Gateway	0.0.0.0	0.0.0.0	Gateway eintragen
Hostname			Hostname eintragen
Domainname			Domainname eintragen
DNS-Server1	0.0.0.0	0.0.0.0	IP-Adresse des ersten DNS-Servers eintragen
DNS-Server2	0.0.0.0	0.0.0.0	Optionale IP-Adresse des zweiten DNS-Servers eintragen
(S)NTP-Server	0.0.0.0	0.0.0.0	IP-Adresse des (S)NTP-Servers eintragen
SNTP Update Time (sec. max. 65535)	0	0	Wartezeit zwischen zwei Anfragen des SNTP-Clients, nach welcher die Netzwerkzeit neu vom (S)NTP-Server angefordert wird, eintragen

10.4 Port

Auf der HTML-Seite „Port“ aktivieren oder deaktivieren Sie die über das IP-Protokoll verfügbaren Dienste.

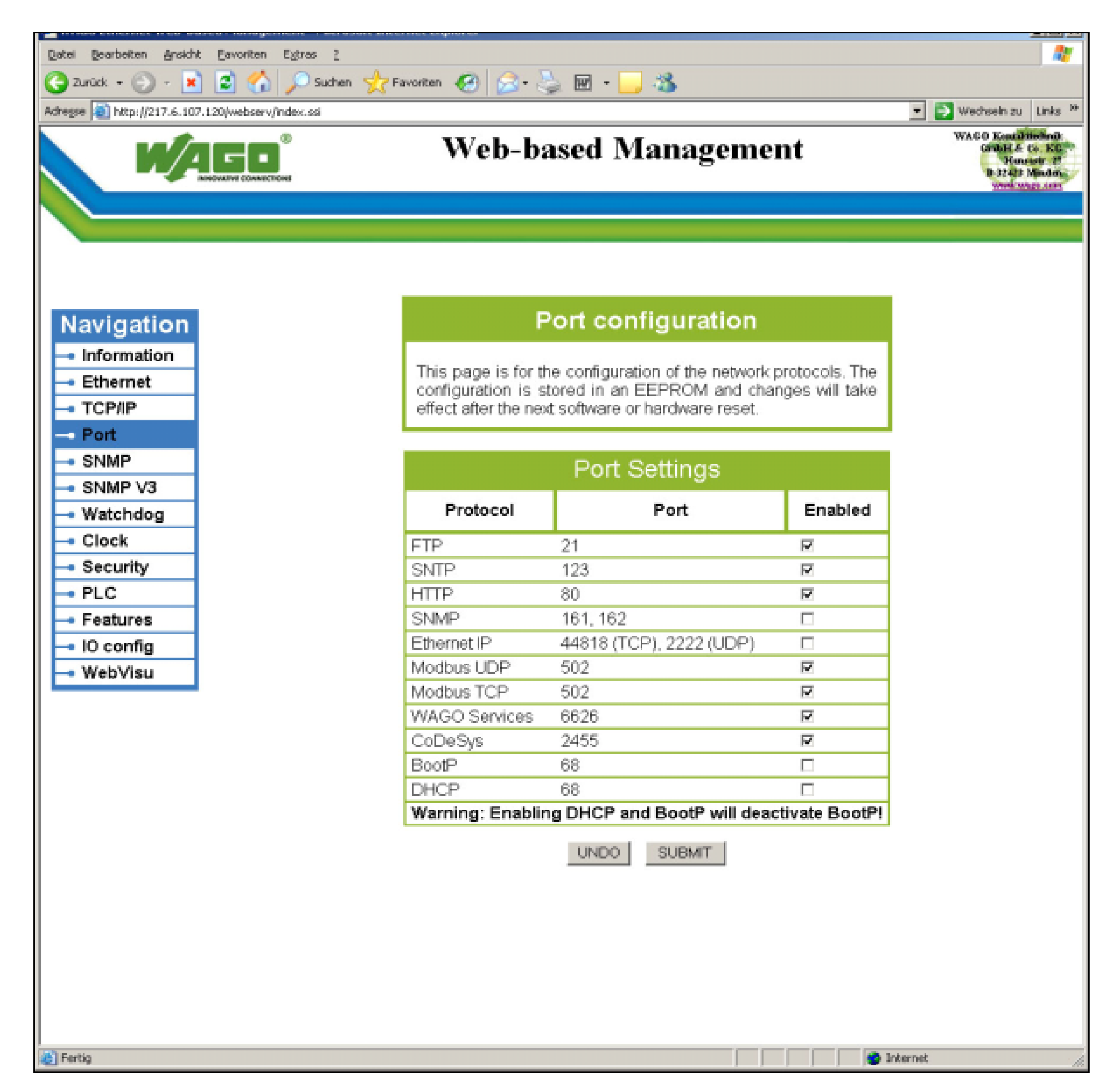
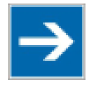


Abbildung 62: WBM-Seite „Port“

Tabelle 44: WBM-Seite „Port“

Port Settings		
Eintrag	**Standardwert**	**Beschreibung**
FTP (Port 21)	Enabled ☑	☑ „File Transfer Protocol“ aktivieren ☐ „File Transfer Protocol“ deaktivieren
SNTP (Port 123)	Enabled ☑	☑ „Simple Network Time Protocol“ aktivieren ☐ „Simple Network Time Protocol“ deaktivieren
HTTP (Port 80)	Enabled ☑	☑ „Hypertext Transfer Protocol“ aktivieren ☐ „Hypertext Transfer Protocol“ deaktivieren
SNMP (Port 161, 162)	Enabled ☐	☑ „Simple Network Management Protocol“ aktivieren ☐ „Simple Network Management Protocol“ deaktivieren
Ethernet IP (TCP-Port 44818, UDP-Port 2222)	Enabled ☐	☑ ETHERNET/IP-Protokoll aktivieren ☐ ETHERNET/IP-Protokoll deaktivieren
Modbus UDP (Port 502)	Enabled ☑	☑ MODBUS/UDP-Protokoll aktivieren ☐ MODBUS/UDP-Protokoll deaktivieren
Modbus TCP (Port 502)	Enabled ☑	☑ MODBUS/TCP-Protokoll aktivieren ☐ MODBUS/TCP-Protokoll deaktivieren
WAGO Services (Port 6626)	Enabled ☑	☑ WAGO-Services (z. B. WAGO-I/O-*CHECK* über Ethernet) aktivieren ☐ WAGO-Services deaktivieren
CoDeSys (Port 2455)	Enabled ☑	☑ CoDeSys aktivieren ☐ CoDeSys deaktivieren
BootP (Port 68)	Enabled ☐	☑ „Boots Trap Protocol“ aktivieren ☐ „Boots Trap Protocol“ deaktivieren
DHCP (Port 68)	Enabled ☐	☑ „Dynamic Host Configuration Protocol“ aktivieren ☐ „Dynamic Host Configuration Protocol“ deaktivieren

Hinweis



DHCP deaktiviert BootP!
Wenn Sie DHCP und BootP gleichzeitig aktivieren, dann wird BootP automatisch deaktiviert.

10.5 SNMP

Auf der HTML-Seite „SNMP“ nehmen Sie Einstellungen für das Simple-Network-Management-Protokoll vor.

SNMP stellt einen Standard für das Management von Geräten in einem TCP/IP-Netzwerk dar. Es dient dem Transport von Kontrolldaten, die den Austausch von Management-Informationen, Status- und Statistikdaten zwischen einzelnen Netzwerkkomponenten und einem Management-System ermöglichen.

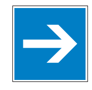
Der Feldbuskoppler/-controller unterstützt SNMP in den Versionen 1, 2c und 3.

In dem Feldbuscontroller umfasst SNMP die allgemeine MIB nach RFC1213 (MIB II).

Zusätzlich ist eine spezielle WAGO-MIB eingebunden.

SNMP wird über den Port 161 abgearbeitet. Die Portnummer für die SNMP-Traps (Meldungen des Agenten) ist 162.

Hinweis



Port 161 und 162 zur Nutzung von SNMP freischalten
Schalten Sie die Ports 161 und 162 im WBM im Menü „Port“ frei, damit der Feldbuskoppler/-controller über SNMP erreichbar ist. Die Portnummern können nicht verändert werden.

Hinweis



Parameter über WBM oder SNMP-Objekte ändern
Die auf den HTML-Seiten einstellbaren Parameter können Sie auch direkt über die entsprechenden SNMP-Objekte verändern.

Information



Weitere Information
Weitere Informationen zu SNMP, zur Management-Information-Base (MIB) und zu Traps (Ereignismeldungen via SNMP) erhalten Sie im Kapitel „Feldbuskommunikation“ > „Kommunikationsprotokolle“ > „SNMP (Simple Network Management Protocol)“.

Betrachten Sie die Einstellungen bezüglich SNMPV1/V2c und SNMPV3 unabhängig voneinander: Die verschiedenen SNMP-Versionen können parallel oder auch einzeln auf einem Feldbuscontroller aktiviert bzw. verwendet werden.



10.5.1 SNMP V1/V2c

Bei SNMP in der Version 1 und 2c handelt es sich um einen Community-Nachrichtenaustausch. Dazu muss der Community-Name der Netzgemeinschaft angegeben werden.

Abbildung 63: WBM-Seite „SNMP“

Tabelle 45: WBM-Seite „SNMP“

SNMP Configuration		
Eintrag	**Wert (Beispiel)**	**Beschreibung**
Name of device	750-841	Gerätename (sysName)
Description	Programmierbarer Feldbuscontroller ETHERNET TCP/IP750-841	Gerätebeschreibung (sysDescription)
Physical location	LOCAL	Standort des Gerätes (sysLocation)
Contact	support@wago.com	E-mail-Kontaktadresse (sysContact)

SNMP v1/v2 Manager Configuration		
Eintrag	**Wert (Beispiel)**	**Beschreibung**
Protocol Enable	SNMP V1/V2c ☑	☑ SNMP-Version 1/2c aktivieren ☐ SNMP-Version 1/2c deaktivieren
Local Community Name	public	verwendeter Community-Name
SNMP v1/v2 Trap Receiver Configuration		
Eintrag	**Wert (Beispiel)**	**Beschreibung**
Trap Receiver 1	0.0.0.0	IP-Adresse des 1. Trap-Empfängers
Community Name 1	public	1. verwendeter Community-Name der Netzgemeinschaft
Trap Version	V1 ◉ V2 ○	V1 ◉ V2 ○ Traps Version 1 aktivieren V1 ○ V2 ◉ Traps Version 2 aktivieren
Trap Receiver 2	0.0.0.0	IP-Adresse des 2. Trap-Empfängers
Community Name 2	public	2. verwendeter Community-Name der Netzgemeinschaft
Trap Version	V1 ◉ V2 ○	V1 ◉ V2 ○ Traps Version 1 aktivieren V1 ○ V2 ◉ Traps Version 2 aktivieren

10.5.2 SNMP V3

In der Version 3 von SNMP ist der Nachrichtenaustausch an Anwender gebunden. Jedes Gerät, welches die über das WBM eingestellten Passwörter kennt, kann Werte aus dem Feldbuskoppler/-controller lesen bzw. schreiben.

Aufgrund seiner Verschlüsselung der Nutzdaten wird SNMP V3 häufig in sicherheitsrelevanten Netzwerken verwendet.



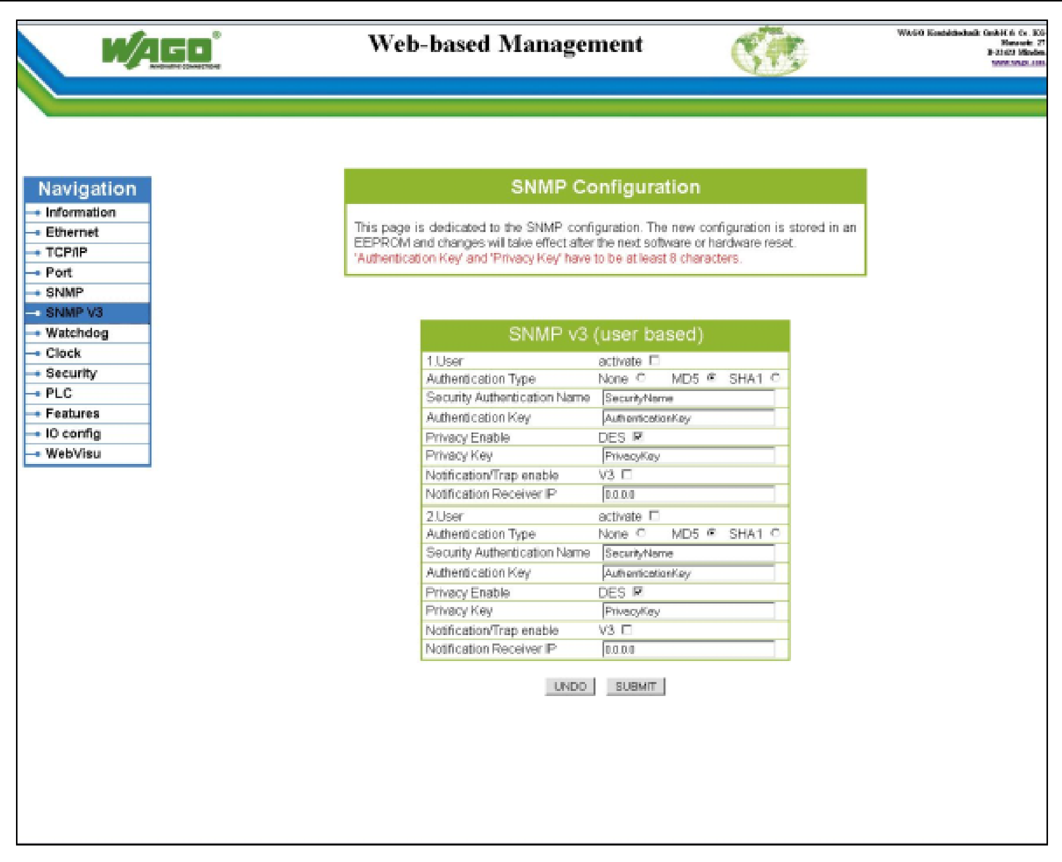


Abbildung 64: WBM-Seite „SNMP V3“

SNMP v3 (user based)		
Eintrag	**Wert (Beispiel)**	**Beschreibung**
1. User / 2. User	activate ☑	☑ Anwender 1 bzw. 2 aktivieren ☐ Anwender 1 bzw. 2 deaktivieren
Authentification Type	None ○ MD5 ◉ SHA1 ○	None ◉ MD5 ○ SHA1 ○ Keine Verschlüsselung der Authentifizierung None ○ MD5 ◉ SHA1 ○ Verschlüsselung der Authentifizierung mit MD5 None ○ MD5 ○ SHA1 ◉ Verschlüsselung der Authentifizierung mit SHA1
Security Authentification Name	Security Name	Name eintragen, wenn „Authentification Type“ MD5 oder SHA1 ausgewählt sind
Authentification Key	Authentification Key	Passwort mit mind. 8 Zeichen eintragen, wenn „Authentification Type“ MD5 oder SHA1 ausgewählt sind
Privacy Enable	DES ☑	☑ DES-Verschlüsselung der Daten aktivieren ☐ DES-Verschlüsselung der Daten deaktivieren
Privacy Key	Privacy Key	Passwort mit mind. 8 Zeichen bei Verschlüsselung mit DES eintragen
Notification/ Trap enable	V3 ☑	☑ Notification-Traps der SNMP-Version 3 aktivieren ☐ Notification-Traps der SNMP-Version 3 deaktivieren
Notification Receiver IP	192.168.1.10	IP-Adresse des Notification-Managers

Über die HTML-Seite „SNMP“ können Sie zwei voneinander unabhängige SNMPV3-Benutzer definieren und aktivieren (User 1 und User 2).



10.6 Watchdog

Auf der HTML-Seite „Watchdog“ nehmen Sie Einstellungen für den Connection- und MODBUS-Watchdog vor.

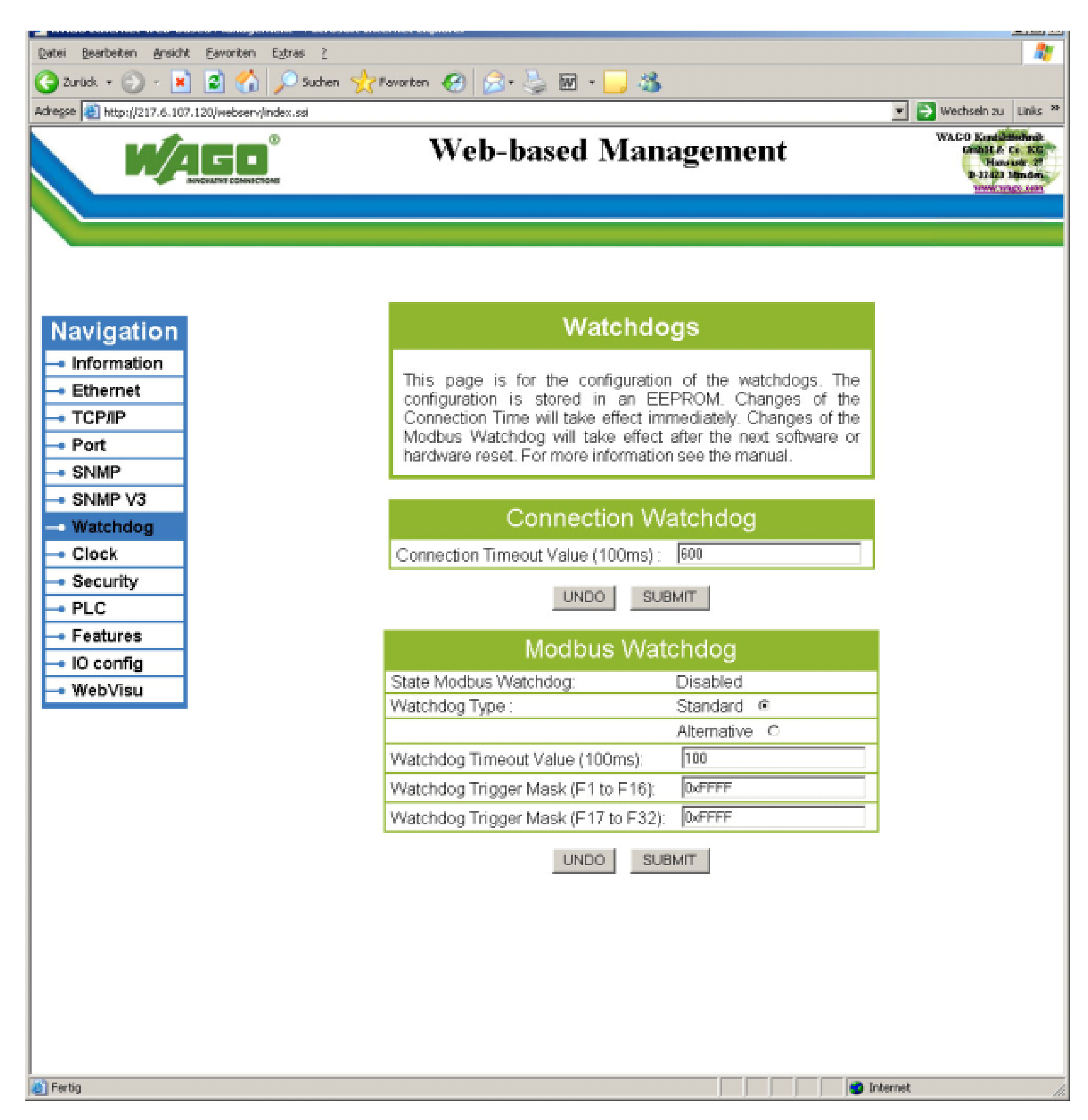


Abbildung 65: WBM-Seite „Watchdog“

Tabelle 46: WBM-Seite „Watchdog“

Connection watchdog		
Eintrag	**Standardwert**	**Beschreibung**
Connection Timeout Value (100 ms)	600	Überwachungszeit für TCP-Verbindungen. Nach Ablauf dieser Zeit ohne erfolgten Datenverkehr wird die TCP-Verbindung geschlossen.
Modbus Watchdog		
Eintrag	**Standardwert**	**Beschreibung**
State Modbus Watchdog	Disabled	„Enabled“ – Watchdog aktiviert “Disabled“ – Watchdog deaktiviert
Watchdog Type	Standard ◉	Die eingestellte Codiermaske (Watchdog Trigger Mask) wird ausgewertet, um zu entscheiden, ob die Watchdog-Zeit zurückzusetzen ist.
	Alternative ○	Mit jedem beliebigen MODBUS/TCP-Telegramm wird die Watchdog-Zeit zurückgesetzt.
Watchdog Timeout Value (100 ms)	100	Überwachungszeit für MODBUS -Verbindungen. Nach Ablauf dieser Zeit ohne empfangenes MODBUS-Telegramm, werden die physikalischen Ausgänge auf '0' gesetzt.
Watchdog Trigger Mask (F 1 to F16)	0xFFFF	Codiermaske für bestimmte MODBUS -Telegramme (Function Code FC1 ... FC16)
Watchdog Trigger Mask (F17 to F32)	0xFFFF	Codiermaske für bestimmte MODBUS -Telegramme (Function Code FC17 ... FC32)



10.7 Clock

Auf der HTML-Seite „Clock“ nehmen Sie Einstellungen für die Feldbuskoppler/-controllerinterne Echtzeituhr vor. Geben Sie hier die aktuelle Uhrzeit und das Datum ein und wählen Sie Winter- oder Sommerzeit aus.

Hinweis



Interne Uhr nach 6 Tagen ohne Spannungsversorgung neu stellen!
Bei der Erstinbetriebnahme oder nach 6 Tagen ohne Spannungsversorgung muss die interne Uhr neu gestellt werden. Erfolgt keine Einstellung, beginnt die Uhr mit dem Datum 01.01.2000 um 0:00 Uhr mit der Zeitmessung.

Hinweis



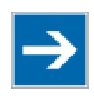
Zur Umstellung von Winter-/Sommerzeit Funktionsblock einbinden!
Die Umstellung zwischen Winter- und Sommerzeit über das Web-based Management-System ist notwendig, wenn Sie die Controller Ihres Netzwerkes über Timeserver synchronisieren. Der Controller selbst unterstützt keine automatische Winter-/Sommerzeitumstellung. Die Umstellung wird über den Funktionsblock „PrgDaylightSaving“ gelöst, den Sie mit der Bibliothek „DaylightSaving.lib“ in der WAGO-I/O-*PRO* CAA einbinden. Fortan erfolgt die Umstellung automatisch, so dass alle Funktionen zeitlich korrekt ausgeführt werden.

Hinweis



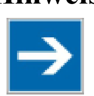
Fehlermeldung in WAGO-I/O-*CHECK* nach Spannungsausfall möglich!
Verwenden Sie nach einem Spannungsausfall die Software WAGO-I/O-*CHECK*“, dann können dort Fehlermeldung auftreten. Rufen Sie in diesem Fall das Web-based Management-System auf und stellen Sie unter „Clock“ die Echtzeit ein. Rufen Sie WAGO-I/O-*CHECK* anschließend erneut auf.

Hinweis



Möglicher Telegrammverlust bei Konfiguration im laufenden Betrieb!
Bei der Konfiguration mittels WAGO-I/O-*CHECK* im laufenden Betrieb kann es zu Telegrammverlusten kommen.

Hinweis



WAGO-RTC-Klemme zur Zeitsynchronisierung nutzen!
Sie können eine WAGO-RTC-Klemme 750-640 in Ihrem Knoten verwenden, um die aktuelle Zeit (Realtime Clock – RTC) in codierter Form in Ihrer übergeordneten Steuerung zu nutzen. Mit dieser RTC-Klemme erreichen Sie eine noch höhere Genauigkeit als mit der koppler- oder controllerinternen Echtzeituhr.



Abbildung 66: WBM-Seite „Clock“

Tabelle 47: WBM-Seite „Clock“

Configuration Data			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
Time on device	Koordinierte Weltzeit UTC	09:16:41	aktuelle Uhrzeit einstellen
Date (YYYY-MM-DD)	Datum abhängig von der UTC	2009-05-06	aktuelles Datum einstellen
Timezone (+/- hour)	0	1 (MEZ)	Zeitzonenabweichung von der koordinierten Weltzeit (UTC) einstellen
Daylight Saving Time (DST) / Summer time	Sommerzeit ☑	Sommerzeit ☑	☑ Sommerzeit aktivieren ☐ Winterzeit aktivieren
12 hour clock	12 Stunden-Uhr ☑	12 Stunden-Uhr ☑	☑ 12-Stunden-Anzeige aktivieren ☐ 24-Stunden- Anzeige aktivieren



10.8 Security

Auf der HTML-Seite „Security“ richten Sie durch Passwörter Lese- und/oder Schreibzugriffe für verschiedene Anwendergruppen zum Schutz vor Konfigurationsänderungen ein.

Hinweis



Passwortänderung nur durch "admin" und nach Software-Reset möglich!
Sie können nur über den Benutzer „admin“ und dem zugehörigen Passwort die Passwörter ändern.
Damit die geänderten Einstellungen wirksam werden, führen Sie mit der Schaltfläche **[Software Reset]** einen Software-Neustart durch.

Hinweis



Passwort-Restriktionen beachten!
Für Passwörter gelten folgende Einschränkungen:

- max.16 Zeichen
- nur Buchstaben und Zahlen
- keine Sonderzeichen und Umlaute



Abbildung 67: WBM-Seite „Security“

Tabelle 48: WBM-Seite „Security“

<table>
<tr><th colspan="4">Webserver Security</th></tr>
<tr><th>Eintrag</th><th>Standardwert</th><th colspan="2">Beschreibung</th></tr>
<tr><td rowspan="2">Webserver authentification enabled</td><td rowspan="2">☑</td><td>☑</td><td>Passwortschutz für den Zugriff auf das Web-Interface aktivieren</td></tr>
<tr><td>☐</td><td>Passwortschutz für den Zugriff auf das Web-Interface deaktivieren</td></tr>
<tr><th colspan="4">Webserver and FTP User configuration *)</th></tr>
<tr><th>Eintrag</th><th>Standardwert</th><th colspan="2">Beschreibung</th></tr>
<tr><td>User</td><td>guest</td><td colspan="2">admin, guest oder user auswählen</td></tr>
<tr><td>Password</td><td></td><td colspan="2">Passwort eintragen</td></tr>
<tr><td>Confirm Password</td><td></td><td colspan="2">Passwort erneut zur Bestätigung eintragen</td></tr>
</table>

*) Standardmäßig sind folgende Gruppen vorgesehen: User: admin Passwort: wago
User: guest Passwort: guest
User: user Passwort: user



Hinweis



Nach Software-Reset Zugriff erneuern!
Wenn Sie auf dieser Seite einen Soft-Reset auslösen, dann startet der Feldbuskoppler/-controller mit den Konfigurationen, die zuvor ins EEPROM geladen wurden, und die Verbindung zum Browser wird unterbrochen.
Haben Sie zuvor die IP-Adresse geändert, müssen Sie mit der geänderten IP-Adresse über den Browser auf das Gerät zugreifen.
Haben Sie die IP-Adresse nicht geändert, sondern andere Einstellungen durchgeführt, können Sie durch Aktualisieren des Browsers die Verbindung wieder herstellen.

10.9 PLC

Auf der HTML-Seite „PLC“ nehmen Sie Einstellungen für die PFC-Funktionalität Ihres Feldbuscontrollers vor.

Web-based Management

Navigation
- Information
- Ethernet
- TCP/IP
- Port
- SNMP
- SNMP V3
- Watchdog
- Clock
- Security
- PLC
- Features
- IO config
- WebVisu

PLC Configuration

This page is for the configuration of the PLC. The configuration is stored in an EEPROM. Changes of the process image setting will take effect after the next software or hardware reset. All other changes will take effect immediately. For more information see the manual.

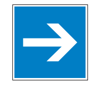
PLC Features

Function	Description	Enabled
Process image	Set outputs to zero, if user program is stopped.	☑
WebVisu	Set 'webvisu.htm' as default.	○
	Open 'webvisu.htm' in frame.	○
	Open 'webvisu.htm' in new window.	◉
I/O configuration	Compatible handling for ea-config.xml	☐
	Insert monitoring entries into ea-config.xml	☑
Target ID	Use alternative Target ID	☐

UNDO SUBMIT

Abbildung 68: WBM-Seite „PLC“

Hinweis



Rückkehr von "WebVisu.htm"-Seite nur über IP-Adresse des Feldbuscontrollers möglich!
Beachten Sie bei Einstellungen für die Seite „WebVisu.htm“, dass diese nicht über Hyperlinks verfügt, die auf die anderen WBM-Seiten verlinken. Um die „WebVisu.htm“ als Startseite zu deaktivieren oder um auf eine der anderen WBM-Seiten zu gelangen, geben Sie in der Adresszeile des Browsers die IP-Adresse Ihres Feldbuscontrollers und die Adresse der ursprünglichen Startseite mit folgender Syntax ein:
http://IP-Adresse Ihres Controllers/webserv/Index.ssi



Tabelle 49: WBM-Seite „PLC“

PLC Features		
Funktion	**Standardwert**	**Beschreibung**
Process image	Set outputs to zero, if user program is stopped ☐	☑ Aktivieren, wenn alle Ausgänge bei Stoppen des Anwenderprogramms auf Null gesetzt werden sollen
		☐ Deaktivieren, wenn alle Werte bei Stoppen des Anwenderprogramms auf dem letzten aktuellen Wert verbleiben sollen
WebVisu	Set 'webvisu.htm' as default ○	◉ Aktivieren, wenn bei einem Aufruf des WMB anstatt der standardmäßigen Startseite „Status Information“ die Seite „Webvisu.htm“ als Startseite geöffnet werden soll
		○ Aktivieren, wenn bei einem Aufruf des WMB die standard-mäßige Startseite „Status Information“ geöffnet werden soll
	Open 'webvisu.htm' in frame ○	◉ Aktivieren, wenn die Seite "WebVisu.htm" in dem selben Fenster geöffnet werden soll.
		○ Aktivieren, wenn die Seite "WebVisu.htm" nicht in dem selben Fenster geöffnet werden soll.
	Open 'webvisu.htm' in new window ◉	◉ Aktivieren, wenn die Seite "WebVisu.htm" in einem neuen Fenster geöffnet werden soll.
		○ Aktivieren, wenn die Seite "WebVisu.htm" nicht in einem neuen Fenster geöffnet werden soll.
I/O configu-ration	Compatible handling for ea-config.xml ☐	☑ Aktivieren, wenn die Schreibberechtigungen auf die Ausgänge aller Busklemmen anhand einer vorhandenen Datei „ea-config.xml“ zugewiesen werden sollen. Beachten Sie dabei, ob bereits eine Steuerungskonfiguration angelegt wurde und, wenn ja, ob diese korrekt oder fehlerhaft ist (siehe nachfolgende Tabelle).
		☐ Deaktivieren, wenn die Schreibberechtigungen auf die Ausgänge aller Busklemmen der SPS zugewiesen werden sollen. Beachten Sie dabei, ob bereits eine Steuerungskonfiguration angelegt wurde und, wenn ja, ob diese korrekt oder fehlerhaft ist (siehe nachfolgende Tabelle).
	Insert monitoring entries into ea-config.xml ☐	☑ Aktivieren, um auf der HTML-Seite „IO config“ für die angezeigten Datenkanäle zusätzlich auch die aktuellen Prozesswerte anzuzeigen.
		☐ Deaktivieren, wenn auf der HTML-Seite „IO config“ keine Prozesswerte angezeigt werden sollen.

	EA-Konfiguration (Funktion aktiviert)	**EA-Konfiguration (Funktion deaktiviert, default)**
Im Projekt ist keine Steuerungskon-figuration angelegt	Die Schreibberechtigungen auf die Ausgänge aller Module werden anhand einer vorhandenen ea-config.xml zugewiesen. Die ea-config.xml muss in jeder Hinsicht fehlerfrei sein, sonst wird dem Standardfeldbus die Schreibberechtigung für alle Module zugewiesen.	Die Ausgänge aller Module werden der SPS zugewiesen. Eine evtl. vorhandene ea-config.xml wird nicht berücksichtigt und überschrieben.
Im Projekt ist eine korrekte Steuerungskon-figuration angelegt	Die Schreibberechtigung auf die Ausgänge der Module wird aus der Steuerungskonfiguration entnommen. Es wird eine entsprechende ea-config.xml im Filesystem erzeugt.	
Im Projekt ist eine falsche Steuerungskon-figuration angelegt	Der Standardfeldbus erhält die Schreibberechtigung auf die Ausgänge aller Module.	



<table>
<tr><td rowspan="2">Target ID</td><td rowspan="2">Use alternative Target ID</td><td rowspan="2">☐</td><td>☑</td><td>Aktivieren, um auf einem Feldbuscontroller ab FW 12 Projekte zu übernehmen und zu verwenden, die mit einer Target-ID bis zur Firmware-Version FW 11 erstellt worden sind.</td></tr>
<tr><td>☐</td><td>Aktivieren, wenn keine Projekte verwendet werden, die mit einer Firmware-Version vor FW 12 erstellt worden sind.</td></tr>
</table>

10.10 Features

Auf der HTML-Seite „Features“ aktivieren bzw. deaktivieren Sie zusätzliche Funktionen.

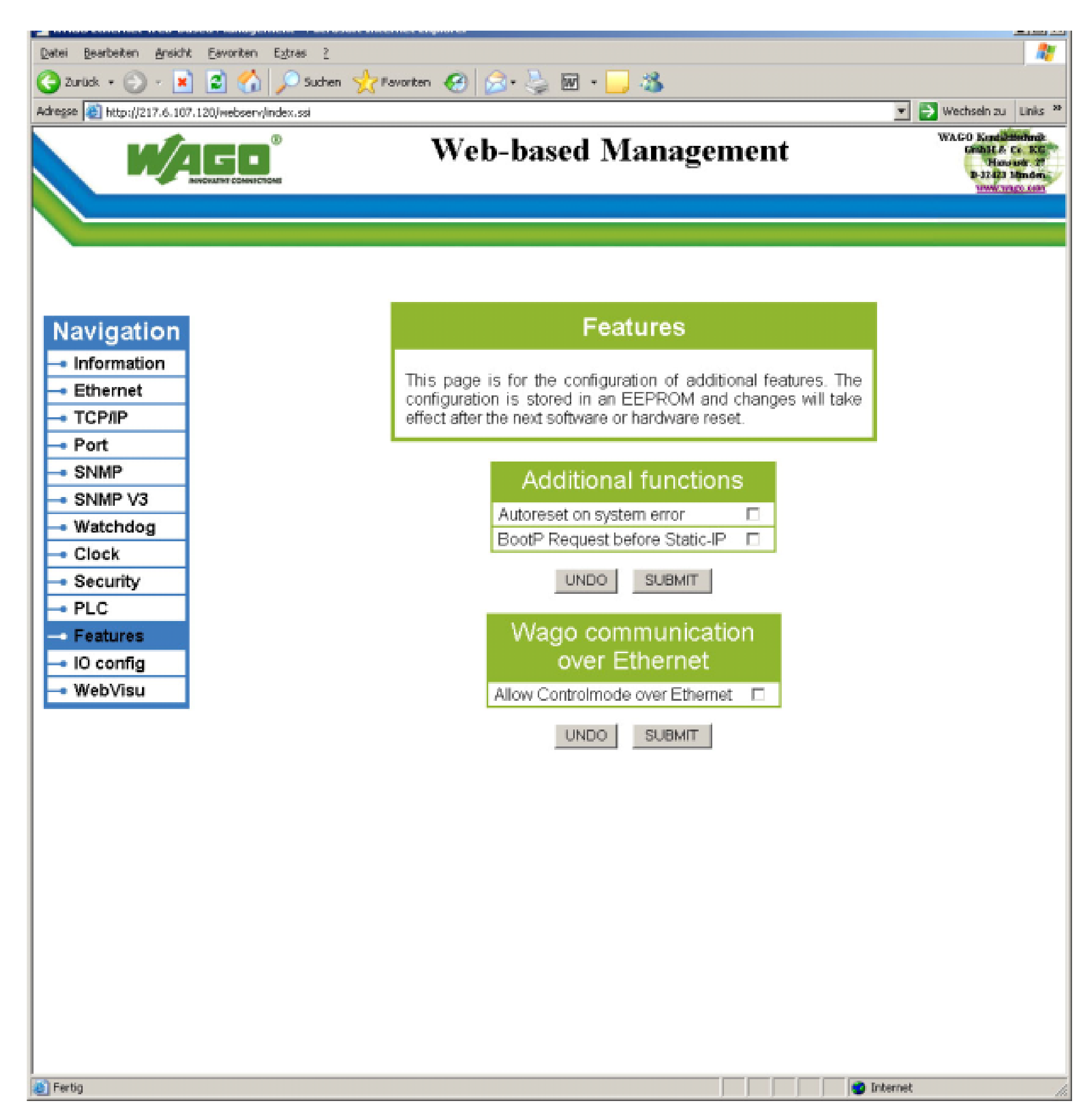


Abbildung 69: WBM-Seite „Features“

Tabelle 50: WBM-Seite „Features“

Eintrag	**Standardwert**	**Beschreibung**
Aditional functions		
Autoreset on system error *)	☐	☑ Automatischer Software-Neustart beim Auftreten eines Systemfehlers aktiv
		☐ Automatischer Software-Neustart beim Auftreten eines Systemfehlers nicht aktiv
BootP Request before Static-IP	☐	☑ Automatisches Setzten der statischen IP-Adressen aktiv. Bei dieser Konfiguration verwendet der Feldbuskoppler/ -controller eine statisch konfigurierte IP-Adresse, falls die Anfrage über BootP fehl schlägt
		☐ Automatisches Setzten der statischen IP-Adressen nicht aktiv. Bei dieser Konfiguration wird die Anfrage der IP-Adresse über BootP im Falle eines Fehlers wiederholt
Wago communication over Ethernet		
Allow Controlmode over Ethernet	☐	☑ Setzten von Ausgangswerten über WAGO-I/O-*CHECK* bei der Kommunikation über ETHERNET aktiv.
		☐ Setzten von Ausgangswerten über WAGO-I/O-*CHECK* bei der Kommunikation über ETHERNET nicht aktiv.

*) Die Funktion "Autoreset on system error" kann bei Verwendung in schwer zugänglichen Bereichen (z. B. im Offshore-Bereich) einen sicheren und schnellen Wiederanlauf bei temporären Fehlern (z. B. bei starken EMV-Störungen) gewährleisten. Der automatische Software-Neustart wird durchgeführt, sobald sich der Feldbuskoppler/-controller in einem Fehlerzustand befindet, der einen Neustart erfordert.
Im Auslieferungszustand ist diese Funktion deaktiviert (Default), so dass beim Auftreten eines Fehlers die Diagnose über den Blinkcode der I/O-LED ausgegeben wird. Nach Fehlerauswertung und -beseitigung ist dann ein manueller Neustart durchzuführen.

10.11 I/O Config

Auf der HTML-Seite „I/O config” sehen Sie eine Übersicht der Konfiguration bzw. der Schreibzugriffsrechte für die Ausgänge Ihres Feldbusknotens.

In dem Fenster wird der Knotenaufbau dargestellt, den Sie mit dem I/O-Konfigurator der WAGO-I/O-*PRO* CAA erstellt haben. Werden keine Busklemmen angezeigt, haben Sie noch keine Hardware-Konfiguration und keine Zuweisung von Schreibzugriffsrechten vorgenommen. In diesem Fall werden entsprechend der Funktion „I/O configuration – Compatible handling for ea-config.xml“ (HTML-Seite „PLC“) die Schreibberechtigungen aller Ausgänge entweder dem Standardfeldbus oder der SPS zugewiesen.

Web-based Management

Navigation: Information, Ethernet, TCP/IP, Port, SNMP, SNMP V3, Watchdog, Clock, Security, PLC, Features, IO config, WebVisu

I/O configuration

Configuration details	
Number of modules on terminalbus	3
Number of modules in I/O configuration	16

I/O configuration file

Pos	Module	Type	Mapping
1	750-4xx	4DI	Fieldbus 1
	M001Ch1		0
	M001Ch2		0
	M001Ch3		0
	M001Ch4		0
2	750-5xx	4DO	Fieldbus 1
	M002Ch1		0
	M002Ch2		0
	M002Ch3		0
	M002Ch4		0
3	750-5xx	8DO	Fieldbus 1
	M003Ch1		0
	M003Ch2		0
	M003Ch3		0
	M003Ch4		0
	M003Ch5		0
	M003Ch6		0
	M003Ch7		0

Abbildung 70: WBM-Seite "I/O config"

Information



Weitere Information
Detaillierte Informationen zu dem I/O-Konfigurator der WAGO-I/O-*PRO* CAA finden Sie im Kapitel „In Betrieb nehmen“.

Ist auf der Webseite „PLC“ zusätzlich noch die Funktion „I/O configuration – Insert monitoring entries into ea-config.xml“ mit einem Haken ausgewählt/aktiviert, werden für die angezeigten Datenkanäle auch die aktuellen Prozesswerte angezeigt.

Tabelle 51: WBM-Seite „I/O configuration“

Configuration details		
Eintrag	**Wert (Beispiel)**	**Beschreibung**
Number of modules on terminalbus	5	Anzahl der Busklemmen (Hardware)
Number of modules in I/O configuration	5	Anzahl der Busklemmen in der Hardware-Konfiguration des I/O-Konfigurators (siehe folgenden Hinweis)

I/O configuration file		
Eintrag	**Wert (Beispiel)**	**Beschreibung**
Pos	1	Position der Busklemme in der Hardware
Module	750-4xx M001Ch1 M001Ch2	Artikelnummer der eingebundenen Busklemme M = Module, 001 = Position 1, Ch1 = Kanal 1 M = Module, 002 = Position 2, Ch2 = Kanal 2
Type	2DI	Busklemmentyp, z. B. 2DI (2-Kanal-Digitaleingangsklemme)
Mapping	Fieldbus 3	Mapping über PLC, Fieldbus 1 etc. (Einträge sind koppler-/controllerabhängig, siehe in WAGO-I/O-*PRO* CAA unter Steuerungsparameter/Modulparameter)

Hinweis



Busklemmen in den I/O-Konfigurator eintragen!
Tragen Sie Ihre verwendeten Busklemmen im I/O-Konfigurator von WAGO-I/O-*PRO* CAA ein. Öffnen Sie dazu im Register **Ressourcen** die **Steuerungskonfiguration** und fügen Sie Ihre Busklemmen der Klemmenbusabbildung hinzu.
Die hinzugefügten Busklemmen müssen in Reihenfolge und Anzahl mit Ihrer Hardware übereinstimmen. Als Kontrolle dienen die Einträge “Number of modules on terminalbus“ und „Number of modules in I/O configuration“ auf der HTML-Seite „PLC“.



10.12 WebVisu

Beim Klicken auf den Link "WebVisu" öffnet sich eine HTML-Seite, auf der die Visualisierung Ihrer programmierten Anwendung angezeigt wird, sofern Sie diese zuvor mit dem Visualisierungseditor in WAGO-I/O-*PRO* CAA bzw. CoDeSys erstellt und in den Feldbuscontroller geladen haben.

Damit bei der Übersetzung Ihres Projektes in WAGO-I/O-*PRO* CAA bzw. CoDeSys automatisch eine HTML-Seite mit Ihrer Visualisierung erstellt wird, nehmen Sie in WAGO-I/O-*PRO* CAA bzw. CoDeSys folgende Einstellungen vor:

1. Öffnen Sie im Register **Ressourcen** die **Zielsystemeinstellungen** mit einem Doppelklick.
2. Öffnen Sie das Register **Visualisierung**.
3. Wählen Sie die Option **Web-Visualisierung** mit einem Haken aus.
4. Bestätigen Sie mit **OK**.

Auf die erstellte WebVisu-HTML-Seite wird von dem Web-based Management-System aus verlinkt.
Dabei können Sie zum Starten dieser HTML-Seite „WebVisu“ verschiedene Einstellungen festlegen:

1. Rufen Sie die Seite „PLC“ im Web-based Management-System auf.
2. a) Um die HTML-Seite „WebVisu“ als Startseite Ihres WBM festzulegen, aktivieren Sie die Option bei der Funktion **WebVisu – Set 'webvisu.htm' as default**.
   Beim Aufruf des Web-based Management-Systems wird dann die „WebVisu“-Seite anstelle der standardmäßigen WBM-Startseite „Information“ geöffnet.
   Die Links zum Wechsel auf die anderen WBM-Seiten stehen jedoch dann nicht mehr zur Verfügung.

Hinweis



Rückkehr von "WebVisu.htm"-Seite nur über IP-Adresse des Feldbuscontrollers möglich!
Die Seite „Webvisu.htm“ verfügt nicht über Hyperlinks, die auf die anderen WBM-Seiten verlinken. Um die „WebVisu.htm“ als Startseite zu deaktivieren oder um auf eine der anderen WBM-Seiten zu gelangen, geben Sie in der Adresszeile des Browsers die IP-Adresse Ihres Feldbuscontrollers und die Adresse der ursprünglichen Startseite mit folgender Syntax ein:
http://IP-Adresse Ihres Controllers/webserv/Index.ssi

b.) Um die HTML-Seite „WebVisu“ in einem Extra-Fenster aufzurufen (Standardeinstellung), aktivieren Sie die Option bei der Funktion **WebVisu – Open 'webvisu.htm' in new window**.
Beim Klicken auf den Link "WebVisu" öffnet sich dann ein neues

Fenster, in dem die HTML-Seite mit der Visualisierung Ihrer programmierten Anwendung angezeigt wird.
Die Links zum Wechsel auf die anderen WBM-Seiten sind bei dieser Einstellung noch verfügbar.

c) Um die HTML-Seite „WebVisu“ direkt auf der WBM-Seite aufzurufen, aktivieren Sie die Option bei der Funktion **WebVisu – Open 'webvisu.htm' in frame**.
Beim Klicken auf den Link "WebVisu" öffnet sich dann direkt in dem WBM-Fenster die HTML-Seite mit der Visualisierung Ihrer programmierten Anwendung in einem Rahmen.
Die Links zum Wechsel auf die anderen WBM-Seiten sind bei dieser Einstellung noch verfügbar.

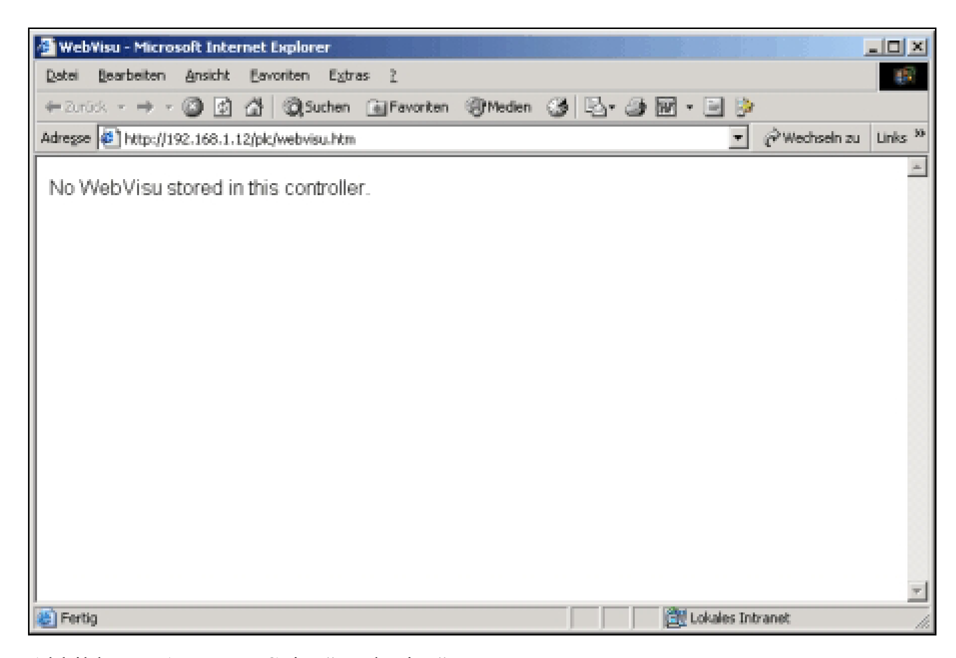


Abbildung 71: WBM-Seite "WebVisu"

11 Diagnose

11.1 LED-Signalisierung

Für die Vor-Ort-Diagnose besitzt der Feldbuscontroller LEDs, die den Betriebszustand des Controllers bzw. des ganzen Knotens anzeigen (siehe folgende Abbildung).

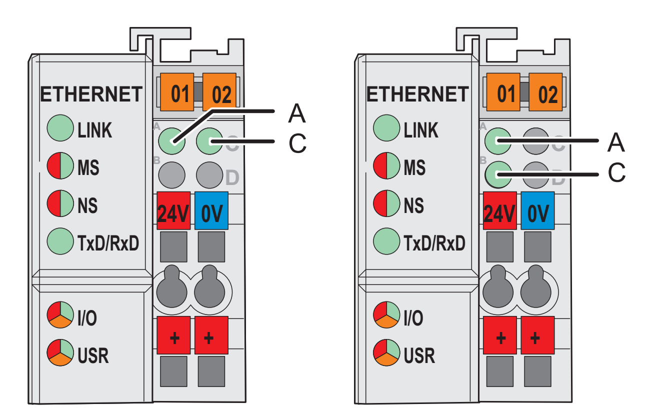


Abbildung 72: Anzeigeelemente (zwei Fertigungsvarianten)

Die Diagnoseanzeigen und deren Bedeutung werden in den nachfolgenden Kapiteln genau erläutert.

Die LEDs sind gruppenweise den verschiedenen Diagnosebereichen zugeordnet:

Tabelle 52: LED-Zuordnung für die Diagnose

Diagnosebereich	LEDs
Feldbusstatus	• LINK • MS • NS • TxD/RxD
Knotenstatus	• I/O • USR
Versorgungsspannungsstatus	• A • B oder C (LED-Position ist fertigungsabhängig)

11.1.1 Feldbusstatus auswerten

Der Betriebszustand der Kommunikation über den Feldbus wird über die obere LED-Gruppe signalisiert, 'LINK', 'MS', 'NS' und 'TxD/RxD'.

Die zweifarbigen LEDs 'MS' (Module Status) und 'NS' (Network Status) werden ausschließlich vom ETHERNET/IP Protokoll verwendet. Die Anzeigen dieser beiden LEDs entsprechen den ETHERNET/IP-Spezifikationen.



Tabelle 53: Diagnose des Feldbusstatus – Abhilfe im Fehlerfall

LED-Status	Bedeutung	Abhilfe
LINK		
grün	Der Feldbusknoten hat Verbindung zu dem physikalischen Netzwerk.	-
aus	Der Feldbusknoten hat keine Verbindung zu physikalischem Netzwerk.	1. Überprüfen Sie das Feldbuskabel.
MS		
grün	Das System arbeitet einwandfrei.	-
grün blinkend	Das System ist noch nicht konfiguriert.	-
rot	Das System zeigt einen nicht behebbaren Fehler an.	1. Führen Sie einen Neustart des Gerätes durch, indem Sie die Versorgungsspannung aus- und einschalten. 2. Sollte der Fehler weiterhin bestehen, wenden Sie sich an den WAGO-I/O-Support.
rot/grün blinkend	Selbsttest	-
aus	Es ist keine Betriebsspannung für das System vorhanden.	1. Überprüfen Sie die Stromversorgung.
NS		
grün	Mindestens eine Verbindung (MODBUS/ TCP oder ETHERNET/IP) ist aufgebaut (auch Verbindung zum Message-Router gilt)	-
grün blinkend	Es besteht keine Verbindung (MODBUS/TCP oder ETHERNET/IP).	-
rot	Das System hat eine doppelt verwendete IP-Adresse erkannt.	1. Verwenden Sie eine noch nicht verwendete IP-Adresse.
rot blinkend	Mindestens eine Verbindung (MODBUS/ TCP oder ETHERNET/IP) hat einen Timeout gemeldet, bei welchem das Gerät als Target fungiert.	1. Starten Sie das Gerät durch Aus- und Einschalten der Versorgungsspannung neu. 2. Bauen Sie die Verbindung erneut auf.
rot/grün blinkend	Selbsttest	-
aus	Dem System ist keine IP-Adresse zugeordnet.	1. Ordnen Sie dem System über BootP, DHCP oder über das Ethernet-Settings-Tool eine IP-Adresse zu.
TxD/RxD		
grün	Datenaustausch über ETHERNET findet statt.	-
aus	Es findet kein Datenaustausch über ETHERNET statt.	-

11.1.2 Knotenstatus auswerten - I/O-LED (Blinkcode-Tabelle)

Der Betriebszustand der Kommunikation zwischen dem Feldbuskoppler/-controller und den Busklemmen wird über die I/O-LED signalisiert.

Tabelle 54: Diagnose des Knotenstatus – Abhilfe im Fehlerfall

LED-Status	Bedeutung	Abhilfe
I/O		
grün	Datenzyklus auf dem Klemmenbus.	Normale Betriebsbedingung
orange blinkend	Der Klemmenbus wird initialisiert. Der Anlauf wird durch ca. 1-2 Sekunden schnelles Blinken angezeigt.	-
rot dauerhaft	Es liegt ein Hardware-Defekt des Feldbuskpplers/-controllers vor.	Tauschen Sie den Feldbuskoppler/-controller aus.
rot blinkend	Blinken mit ca. 10 Hz weist auf einen allgemeinen Klemmenbusfehler hin.	Beachten Sie nachfolgenden Blinkcode.
rot zyklisch blinkend	Es werden auftretende Klemmenbusfehler mit bis zu drei nacheinander folgende Blinksequenzen angezeigt. Zwischen diesen Sequenzen ist jeweils eine kurze Pause.	Werten Sie die angezeigten Blink-sequenzen anhand der nachfolgenden Blinkcode-Tabelle aus. Das Blinken zeigt eine Fehlermeldung an, die sich aus einem Fehlercode und einem Fehlerargument zusammensetzt.
aus	Kein Datenzyklus auf dem Klemmenbus.	Die Versorgungsspannung des Feldbuskopplers/-controllers ist nicht eingeschaltet.

Nach Einschalten der Versorgungsspannung läuft das Gerät hoch. Dabei leuchtet die I/O-LED orange.

Nach fehlerfreiem Hochlauf zeigt die I/O-LED grünes Dauerlicht.
Im Fehlerfall blinkt die I/O-LED rot.

Mit Hilfe eines Blinkcodes werden detaillierte Fehlermeldungen angezeigt. Ein Fehler wird über bis zu 3 Blinksequenzen zyklisch dargestellt.

Nach Beseitigung eines Fehlers ist der Feldbusknoten durch Aus- und Einschalten der Versorgungsspannung des Gerätes neu zu starten.



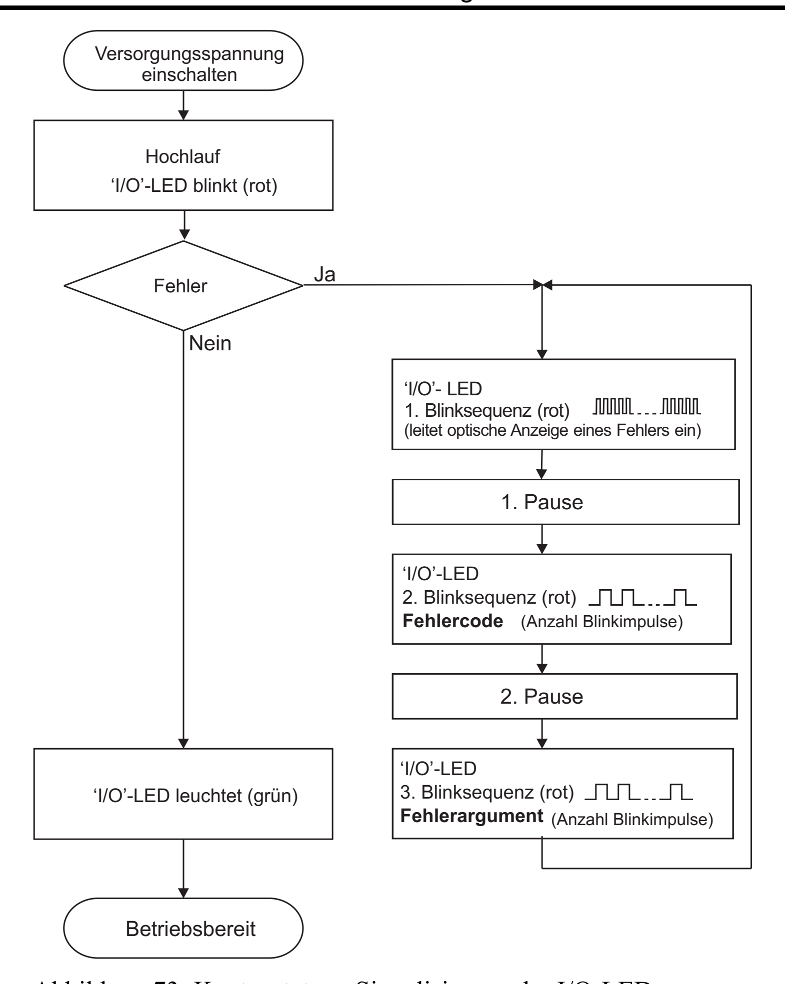


Abbildung 73: Knotenstatus - Signalisierung der I/O-LED

1. Blinksequenz (ca. 10 Hz)	**Pause**	**2. Blinksequenz** (ca 1 Hz)	**Pause**	**3. Blinksequenz** (ca 1 Hz)
(Einleitung der Fehlermeldung)		**Fehlercode x** (x = Anzahl der Blinkimpulse)		**Fehlerargument y** (y = Anzahl der Blinkimpulse)

Abbildung 74: Codierung der Fehlermeldung

Beispiel eines Klemmenfehlers:

- Die I/O-LED leitet mit der 1. Blinksequenz (ca. 10 Hz) die Fehleranzeige ein.
- Nach der ersten Pause folgt die 2. Blinksequenz (ca. 1 Hz):
  Die I/O-LED blinkt viermal.
  Damit wird der Fehlercode 4 "Datenfehler Klemmenbus" signalisiert.
- Nach der zweiten Pause folgt die 3. Blinksequenz (ca. 1 Hz):
  Die I/O-LED blinkt zwölf mal.
  Das Fehlerargument 12 bedeutet, dass der Klemmenbus nach der 12. Busklemme unterbrochen ist.

Somit ist die 13. Busklemme entweder defekt oder aus dem Verbund herausgezogen.

Tabelle 55: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 1

Fehlercode 1: "Hardware- und Konfigurationsfehler"		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
1	Interner Speicherüberlauf bei Inlinecode-Generierung.	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der Busklemmen. 3. Schalten Sie die Versorgungsspannung wieder ein. 4. Sollte der Fehler weiterhin bestehen, tauschen Sie den Feldbuscontroller aus.
2	Busklemme(n) mit nicht unterstützter Datenstruktur	1. Ermitteln Sie die fehlerhafte Busklemme, indem Sie die Versorgungsspannung ausschalten. 2. Stecken sie die Endklemme in die Mitte des Knotens. 3. Schalten Sie die Versorgungsspannung wieder ein. 4. --- Blinkt die LED weiter? --- Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der ersten Hälfte des Knotens (zum Feldbuscontroller hin). --- Blinkt die LED nicht? --- Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der zweiten Hälfte des Knotens (vom Feldbuscontroller weg). 5. Schalten Sie die Versorgungsspannung wieder ein. 6. Wiederholen Sie den im Schritt 4 beschriebenen Vorgang mit halbierten Schrittweiten, bis die fehlerhafte Busklemme gefunden ist. 7. Tauschen Sie die fehlerhafte Busklemme aus. 8. Erkundigen Sie sich nach einem Firmware-Update für den Feldbuscontroller.
3	Ungültige Prüfsumme im Parameterbereich des Feldbuscontrollers	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
4	Fehler beim Schreiben in das serielle EEPROM	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
5	Fehler beim Lesen aus dem seriellen EEPROM	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
6	Die ermittelte Bus-klemmen-Konfigu-ration nach einem Klemmenbus-Reset (AUTORESET) differiert zu der, die beim letzten Hoch-lauf des Feldbus-controllers ermittelt wurde.	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu.



Tabelle 55: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 1

Fehlercode 1: "Hardware- und Konfigurationsfehler"		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
7	Ungültige Hardware-Firmware-Kombination	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
8	Zeitüberschreitung beim Zugriff auf das serielle EEPROM	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
9	Buscontroller Initialisierungsfehler	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
10	Pufferspannungs-ausfall Echtzeituhr (RTC)	1. Stellen Sie die Uhr. 2. Erhalten Sie die Versorgungsspannung des Feldbuscontrollers für mindestens 15 Minuten zwecks Aufladung des Goldcaps aufrecht.
11	Fehler beim Lesezugriff auf die Echtzeituhr (RTC)	1. Stellen Sie die Uhr. 2. Erhalten Sie die Versorgungsspannung des Feldbuscontrollers für mindestens 15 Minuten zwecks Aufladung des Goldcaps aufrecht.
12	Fehler beim Schreibzugriff auf die Echtzeituhr (RTC)	1. Stellen Sie die Uhr. 2. Erhalten Sie die Versorgungsspannung des Feldbuscontrollers für mindestens 15 Minuten zwecks Aufladung des Goldcaps aufrecht.
13	Fehler Uhren-Interrupt	1. Stellen Sie die Uhr. 2. Erhalten Sie die Versorgungsspannung des Feldbuscontrollers für mindestens 15 Minuten zwecks Aufladung des Goldcaps aufrecht.
14	Maximale Anzahl an Gateway- bzw. Mailbox-Busklemmen überschritten	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der entsprechenden Busklemmen auf ein zulässiges Maß. 3. Schalten Sie die Versorgungsspannung wieder ein.

Tabelle 56: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 2

Fehlercode 2: -nicht genutzt-		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
-	nicht genutzt	-

Tabelle 57: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 3

Fehlercode 3: "Protokollfehler Klemmenbus"		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
-	Klemmenbus-kommunikation gestört, fehlerhafte Baugruppe ist nicht identifizierbar	--- Befinden sich Potentialeinspeiseklemmen mit Busnetzteil (750-613) im Knoten? --- 1. Überprüfen Sie, ob diese Klemmen korrekt mit Spannung versorgt werden. 2. Entnehmen Sie dieses dem Zustand der zugehörigen Status-LEDs. --- Sind alle Klemmen ordnungsgemäß angeschlossen oder befinden sich keine Busklemmen vom Typ 750-613 im Knoten? --- 1. Ermitteln Sie die fehlerhafte Busklemme, indem Sie die Versorgungsspannung ausschalten. 2. Stecken sie die Endklemme in die Mitte des Knotens. 3. Schalten Sie die Versorgungsspannung wieder ein. 4. --- Blinkt die LED weiter? --- Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der ersten Hälfte des Knotens (zum Feldbuscontroller hin). --- Blinkt die LED nicht? --- Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der zweiten Hälfte des Knotens (vom Feldbuscontroller weg). 5. Schalten Sie die Versorgungsspannung wieder ein. 6. Wiederholen Sie den im Schritt 4 beschriebenen Vorgang mit halbierten Schrittweiten, bis die fehlerhafte Busklemme gefunden ist. 7. Tauschen Sie die fehlerhafte Busklemme aus. 8. Befindet sich nur noch eine Busklemme am Feldbus-controller und die LED blinkt, ist entweder diese Klemme defekt oder der Feldbuscontroller. Tauschen Sie die defekte Komponente.



Tabelle 58: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 4

Fehlercode 4: "Physikalischer Fehler Klemmenbus"		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
-	Fehler bei der Klemmenbus-datenübertragung oder Unterbrechung des Klemmenbusses an dem Feldbuscontroller	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Stecken Sie eine Busklemme mit Prozessdaten hinter den Feldbuscontroller. 3. Schalten Sie die Versorgungsspannung ein. 4. Beobachten Sie das signalisierte Fehlerargument. - Wird kein Fehlerargument auf der I/O-LED ausgegeben? - 5. Tauschen Sie den Feldbuscontroller aus. - Wird ein Fehlerargument auf der I/O-LED ausgegeben? - 5. Ermitteln Sie die fehlerhafte Busklemme, indem Sie die Versorgungsspannung ausschalten. 6. Stecken sie die Endklemme in die Mitte des Knotens. 7. Schalten Sie die Versorgungsspannung wieder ein. 8. - Blinkt die LED weiter? - Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der ersten Hälfte des Knotens (zum Feldbuscontroller hin). - Blinkt die LED nicht? - Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der zweiten Hälfte des Knotens (vom Feldbuscontroller weg). 9. Schalten Sie die Versorgungsspannung wieder ein. 10. Wiederholen Sie den im Schritt 6 beschriebenen Vorgang mit halbierten Schrittweiten, bis die fehlerhafte Busklemme gefunden ist. 11. Tauschen Sie die fehlerhafte Busklemme aus. 12. Befindet sich nur noch eine Busklemme am Feldbus-controller und die LED blinkt, ist entweder diese Klemme defekt oder der Feldbuscontroller. Tauschen Sie die defekte Komponente.
n*	Es liegt eine Klemmenbus-unterbrechung hinter der n-ten Busklemme mit Prozessdaten vor.	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie die (n+1)-te Busklemme mit Prozessdaten aus. 3. Schalten Sie die Versorgungsspannung ein.

* Die Anzahl der Blinkimpulse (n) zeigt die Position der Busklemme an.
Busklemmen ohne Daten werden nicht mitgezählt (z. B. Einspeiseklemme ohne Diagnose)

Tabelle 59: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 5

Fehlercode 5: "Initialisierungsfehler Klemmenbus"		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
n*	Fehler bei der Registerkom-munikation während der Klemmenbus-Initialisierung	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie die (n+1)-te Busklemme mit Prozessdaten aus. 3. Schalten Sie die Versorgungsspannung ein.

* Die Anzahl der Blinkimpulse (n) zeigt die Position der Busklemme an.
Busklemmen ohne Daten werden nicht mitgezählt (z. B. Einspeiseklemme ohne Diagnose)

Tabelle 60: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 6

Fehlercode 6: " Projektierungsfehler Knotenkonfiguration"		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
1	Ungültige MAC-ID	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung ein.
2	Initialisierungsfehler ETHERNET-Hardware	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Wird der Fehler weiterhin gemeldet? Tauschen Sie den Feldbuscontroller aus.
3	Initialisierungsfehler TCP/IP-Stack	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Wird der Fehler weiterhin gemeldet? Tauschen Sie den Feldbuscontroller aus.
4	Konfigurationsfehler Netzwerk (keine IP-Adresse)	1. Überprüfen Sie die Einstellungen des BootP-Servers.
5	Fehler bei der Initialisierung eines Applikationsproto-kolls	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Wird der Fehler weiterhin gemeldet? Tauschen Sie den Feldbuscontroller aus.
6	Maximale Prozessabbildgröße überschritten	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der Busklemmen.
7	IP-Adresse des Feldbuscontrollers ist mehrfach im Netzwerk vorhanden	1. Ändern Sie die Konfiguration: Verwenden Sie eine noch nicht im Netz vorhandene IP-Adresse.. 2. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu.
8	Fehler beim Erstellen des Prozessabbildes	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der Busklemmen. 3. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 4. Wird der Fehler weiterhin gemeldet? Tauschen Sie den Feldbuscontroller aus.

Tabelle 61: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 7...9

Fehlercode 7... 9 – nicht genutzt –		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
-	nicht genutzt	-



Tabelle 62: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 10

Fehlercode 10: "Fehler bei der SPS-Programmbearbeitung"		
Fehler-argument	**Fehlerbeschreibung**	**Abhilfe**
1	Fehler beim Aufsetzen des PFC-Laufzeitsystems	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Sollte der Fehler weiterhin gemeldet werden, wenden Sie sich an den I/O-Support.
2	Fehler beim Generieren des PFC-Inline-Codes	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Sollte der Fehler weiterhin gemeldet werden, wenden Sie sich an den I/O-Support.
3	Eine IEC-Task hat die maximale Laufzeit überschritten oder das Aufrufintervall der IEC-Task konnte nicht eingehalten werden (Zeitüberwachung)	1. Überprüfen Sie die Task-Konfiguration bezüglich der eingestellten Aufrufintervalle und Überwachungszeiten.
4	Fehler beim Initialisieren der PFC Web-Visualisierung	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Sollte der Fehler weiterhin bestehen, führen Sie in WAGO-I/O-*PRO* CAA einen Reset (Ursprung) durch. 3. Übersetzen Sie das Projekt erneut. 4. Bringen Sie das Projekt wieder auf den Feldbuscontroller.
5	Fehler beim Abgleich der Steuerungs-konfiguration mit dem Klemmenbus	1. Überprüfen Sie die Angabe der gesteckten Klemmen in der CoDeSys-Steuerungskonfiguration 2. Gleichen Sie diese mit den tatsächlich gesteckten Klemmen ab. 3. Übersetzen Sie das Projekt erneut. 4. Bringen Sie das Projekt wieder auf den Feldbuscontroller.

Tabelle 63: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 11

Fehlercode 11: "Gateway-/Mailbox-Klemmen Fehler"		
Fehler-argument	**Fehlerbeschreibung**	**Abhilfe**
1	Es sind zu viele Gateway-Klemmen gesteckt	1. Vermindern Sie die Zahl der Gateway-Klemmen
2	Maximale Mailbox-Größe überschritten	1. Verkleinern Sie die Mailbox-Größe
3	Maximale PA-Größe überschritten aufgrund von gesteckten Gateway-Klemmen	1. Verkleinern Sie die Datenbreite der Gateway-Klemmen

* Die Anzahl der Blinkimpulse (n) zeigt die Position der Busklemme an.
Busklemmen ohne Daten werden nicht mitgezählt (z. B. Einspeiseklemme ohne Diagnose)

11.1.2.1 USR-LED

Für die visuelle Ausgabe von Informationen steht dem Anwender die unterste Anzeige LED („USR“) zur Verfügung.

Die Ansteuerung der LED aus dem Anwenderprogramm erfolgt mit den Funktionen aus der WAGO-I/O-*PRO*-CAA-Bibliothek „Visual.lib“.

11.1.3 Versorgungsspannungsstatus auswerten

Im Einspeiseteil des Gerätes befinden sich zwei grüne LEDs zur Anzeige der Versorgungsspannung.
Die LED „A“ zeigt die 24V-Versorgung des Feldbusknotens an.
Die LED „B“ bzw. „C“ meldet die Versorgung, die an den Leistungskontakten für die Feldseite zur Verfügung steht.

Tabelle 64: Diagnose des Versorgungsspannungsstatus – Abhilfe im Fehlerfall

LED-Status	Bedeutung	Abhilfe
A		
grün	Die Betriebsspannung für das System ist vorhanden.	-
aus	Es ist keine Betriebsspannung für das System vorhanden.	Überprüfen Sie die Versorgungsspannung für das System (24 V und 0 V).
B oder C		
grün	Die Betriebsspannung für die Leistungskontakte ist vorhanden.	-
aus	Es ist keine Betriebsspannung für die Leistungskontakte vorhanden.	Überprüfen Sie die Versorgungs-spannung für die Leistungskontakte (24 V und 0 V).



11.2 Fehlerverhalten

11.2.1 Feldbusausfall

Ein Feldbus- und damit ein Verbindungsausfall liegt vor, wenn die eingestellte Reaktionszeit des Watchdogs ohne Anstoß durch die übergeordnete Steuerung abgelaufen ist. Dies kann beispielsweise passieren, wenn der Master abgeschaltet oder das Buskabel unterbrochen ist. Auch ein Fehler im Master kann zum Feldbusausfall führen. Es ist keine Verbindung über ETHERNET gegeben.

Der MODBUS-Watchdog überwacht die über das MODBUS-Protokoll laufende MODBUS-Kommunikation. Sofern der MODBUS-Watchdog konfiguriert und aktiviert wurde, wird ein Feldbusausfall durch das Leuchten der roten I/O-LED angezeigt.

Eine protokollunabhängige Feldbusüberwachung ist über den Funktionsblock 'FBUS_ERROR_INFORMATION' der Bibliothek 'Mod_com.lib' möglich, der die physikalische Verbindung zwischen Busklemmen und Feldbuscontroller überprüft und die Auswertung der Watchdog-Register im Steuerungsprogramm übernimmt. Der Klemmenbus bleibt funktionsfähig und die Prozessabbilder bleiben erhalten. Das Steuerungsprogramm kann autark abgearbeitet werden.

FBUS_ERROR_INFORMATION
FBUS_ERROR
ERROR

Abbildung 75: Funktionsblock zur Ermittlung des Feldbusausfalls

'FBUS_ERROR' (BOOL)	= FALSE	= kein Fehler
	= TRUE	= Feldbusausfall
'ERROR' (WORD)	= 0	= kein Fehler
	= 1	= Feldbusausfall

Mit Hilfe dieser Funktionsblockausgänge und einem entsprechend programmierten Steuerungsprogramm kann der Knoten bei Feldbusausfall in einen sicheren Zustand geführt werden.

Information



Feldbusausfallerkennung über das MODBUS-Protokoll:
Detaillierte Informationen zu dem Watchdog-Register entnehmen Sie dem Kapitel „MODBUS-Funktionen“, „Watchdog (Verhalten bei Feldbusausfall)“.
Protokollunabhängige Feldbusausfall-Erkennung:
Die Bibliothek 'Mod_com.lib' mit dem Funktionsblock 'FBUS_ERROR_INFORMATION' ist standardmäßig im Setup der WAGO-I/O-*PRO* CAA enthalten. Sie binden die Bibliothek über das Register „Ressourcen“ links unten auf der Arbeitsfläche ein. Klicken Sie auf **Einfügen** und **weitere Bibliotheken**. Die Mod_com.lib befindet sich im Ordner C:\Programme\WAGO Software\CoDeSys V2.3\Targets\WAGO\Libraries\32_Bit

11.2.2 Klemmenbusfehler

Ein Klemmenbusfehler wird über die I/O-LED angezeigt.

I/O-LED blinkt rot:
Bei einem Klemmenbusfehler erzeugt der Feldbuscontroller eine Fehlermeldung (Fehlercode und Fehlerargument).
Ein Klemmenbusfehler entsteht beispielsweise durch eine herausgezogene Busklemme.
Wenn dieser Fehler während des Betriebes auftritt, verhalten sich die Ausgangsklemmen wie beim Klemmenbusstopp.
Wenn der Klemmenbusfehler behoben ist, läuft der Feldbuscontroller nach einem Aus- und Einschalten wie beim Betriebsstart hoch. Die Übertragung der Prozessdaten wird wieder aufgenommen und die Ausgänge im Knoten werden entsprechend gesetzt.

Soll in dem Steuerungsprogramm der Funktionsbaustein 'KBUS_ERROR_INFORMATION' ausgewertet werden, dann sind die Ausgangswerte 'ERROR','BITLEN', 'TERMINALS' und 'FAILADDRESS' relevant.

'ERROR'	= FALSE	= kein Fehler
	('BITLEN'	= Bitlänge des Klemmenbus-Schieberegisters
	'TERMINALS'	= Anzahl der gesteckten Busklemmen)
'ERROR'	= TRUE	= Klemmenbusfehler
	('BITLEN'	= 0
	'TERMINALS'	= 0)
	'FAILADRESS'	= Position der Busklemme, nach der die Klemmenbusunterbrechung aufgetreten ist, analog zu dem ausgeblinkten Fehlerargument der I/O-LED)



12 Feldbuskommunikation

Die Feldbuskommunikation zwischen Master-Anwendung und einem auf dem ETHERNET-Standard basierenden WAGO-Feldbuskoppler/-controller findet in der Regel über ein feldbusspezifisch implementiertes Anwendungsprotokoll statt.

Je nach Anwendung, kann dieses z. B. MODBUS/TCP (UDP), EtherNet/IP, BACnet/IP, KNXnet/IP, PROFINET, SERCOS III oder sonstiges sein.

Hinzu kommen zu dem ETHERNET-Standard und dem feldbusspezifischen Anwendungsprotokoll außerdem noch einige, für eine zuverlässige Kommunikation und Datenübertragung wichtige Kommunikationsprotokolle und darauf aufbauend noch weitere Protokolle für die Konfiguration und Diagnose des Systems, die in den ETHERNET basierenden WAGO-Feldbuskoppler/-controller implementiert sind.

Diese Protokolle werden in den weiteren Kapiteln näher erläutert.

12.1 Allgemeine ETHERNET-Informationen

ETHERNET ist eine Technologie, die sich für die Datenübertragung in der Informationstechnik und in der Bürokommunikation hervorragend bewährt und etabliert hat. Auch in dem privaten PC Bereich ist ETHERNET in kürzester Zeit weltweit der Durchbruch gelungen.

Diese Technologie wurde 1979 gemeinsam von den Firmen Xerox, INTEL und DEC als Spezifikation für ein lokales Netzwerk (LAN) entwickelt. Eine Normung (IEEE 802.3) fand im Jahre 1983 statt.

Als Übertragungsmedium benutzt ETHERNET überwiegend Koaxialkabel oder verdrillte Zweidrahtleitungen. Die Komponenten sind fast überall erhältlich und sehr preiswert. Eine Anbindung an oft schon vorhandene Netze (LAN, Internet) kann problemlos realisiert werden, und der Datenaustausch ist mit einer Übertragungsrate von 10 Mbit/s oder bei einigen Feldbuskopplern/-controllern auch 100 MBit/s sehr schnell.

Für die Kommunikation zwischen verschiedenen Systemen wurde ETHERNET zusätzlich zur Norm IEEE 802.3 mit einer übergeordneten Kommunikationssoftware ausgerüstet, mit TCP/IP (Transmission Control Protocol / Internet Protocol). Der TCP/IP-Protokollstack bietet eine hohe Zuverlässigkeit bei der Informationsübertragung.

In den von WAGO entwickelten Feldbuskopplern/-controllern, die auf ETHERNET basieren, ist auf der Basis des TCP/IP-Stacks eine Vielzahl von Applikationsprotokollen implementiert.

Diese Protokolle ermöglichen dem Anwender Anwendungen (Master-Applikationen) durch standardisierte Schnittstellen zu erstellen und Prozessdaten über eine ETHERNET-Schnittstelle zu übermitteln.

Neben einer Reihe von Konfigurations- und Diagnose-Protokollen, sind in der Regel für die Steuerung der Daten von den I/O-Modulen bzw. Busklemmen, je nach Feldbuskoppler/-controller, feldbusspezifische Protokolle implementiert. Dieses sind z. B. MODBUS/TCP (UDP), EtherNet/IP, BACnet, KNXnet/IP, PROFINET, SERCOS III, oder sonstige.

Informationen, wie den Aufbau des Feldbusknoten, Netzstatistiken und Diagnoseinformationen, sind in den ETHERNET Feldbuskopplern/-controllern gespeichert und können als HTML-Seiten über einen Web Browser (Microsoft Internet-Explorer, Netscape Navigator,..) direkt aus diesen Seiten ausgelesen werden.

Darüber hinaus können, je nach industrieller Anforderung, über das Web-based Management-System verschiedene Einstellungen, wie z. B. die Auswahl der Protokolle, TCP/IP-, Uhrzeit- und Sicherheitskonfigurationen, vorgenommen werden.

Für Feldbuskoppler/-controller, die ein internes Filesystem besitzen, können über FTP aber auch eigens erstellte Webseiten in die Feldbuskoppler/-controller geladen werden.

12.1.1 Netzwerkaufbau - Grundlagen und Richtlinien

Für den Aufbau eines einfachen ETHERNET Netzwerkes benötigen Sie einen PC mit Netzwerkkarte, ein Verbindungskabel, einen ETHERNET Feldbusknoten und ein DC 24 V Netzgerät für die Spannungsversorgung.

Jeder Feldbusknoten besteht aus einem Feldbuskoppler/-controller, und einer Anzahl entsprechend benötigter I/O-Module bzw. Busklemmen.

An die digitalen oder analogen Ein- und Ausgänge werden auf der Feldseite Sensoren und Aktoren angeschlossen. Über diese werden die Prozesssignale erfasst bzw. können Signale an den Prozess ausgegeben werden.

Der WAGO-ETHERNET-Feldbusknoten benötigt, außer einem PC mit Netzwerkkarte, keine zusätzlichen Master-Komponenten und kann somit mit dem Feldbus-Anschluss problemlos an lokale oder globale Netze angebunden werden. Weiterhin können die aus der Rechnervernetzung bekannten Komponenten wie Hubs, Switches oder Repeater eingesetzt werden.

Durch die Verwendung von ETHERNET als Feldbus, wird eine durchgängige Datenübertragung zwischen Fertigung und Büro geschaffen. Bei Anschluss des ETHERNET Feldbusknoten an das Internet, können sogar weltweit, bei Bedarf auch von mehreren Stationen gleichzeitig, industrielle Prozessdaten für alle Arten von Applikationen abgerufen werden. Somit wird eine standortunabhängige Überwachung, Visualisierung, Fernwartung und Steuerung von Prozessen ermöglicht.

Die Feldbuskommunikation zwischen Master-Anwendung und Feldbuskoppler/-controller findet dann über das feldbusspezifisch implementierte



Anwendungsprotokoll statt, also z. B. über MODBUS/TCP (UDP), EtherNet/IP, BACnet, KNXnet/IP, PROFINET, Powerlink, Sercos III oder sonstiges.

12.1.1.1 Übertragungsmedien

Allgemeine ETHERNET Übertragungsstandards

Zur Übertragung von Daten unterstützt der ETHERNET-Standard zahlreiche Technologien, die sich in verschiedenen Kenngrößen, wie z.B. Übertragungsgeschwindigkeit, Medium, Segmentlänge und Übertragungsart, unterscheiden.

Tabelle 65: ETHERNET-Übertragungsstandards

1Base5	verwendet ein 24 AWG UTP (Verdrilltes Adernpaar) für ein 1Mbit/s Basisbandsignal für Entfernungen bis zu 500 m (250 m pro Segment) in einer physischen Stern-Topologie.
10Base2	verwendet ein 5 mm 50 Ohm Koaxialkabel für ein 10 Mbit/s Basisbandsignal für Entfernungen bis zu 185 m in einer physischen Bus-Topologie (oft als Thin ETHERNET, ThinNet oder Cheapernet bezeichnet).
10Base5	verwendet ein 10 mm 50 Ohm Koaxialkabel für 10 Mbit/s ein Basisbandsignal für Entfernungen bis zu 500 m in einer physischen Bus-Topologie (oft als Thick ETHERNET bezeichnet).
10Base-F	verwendet ein Glasfaserkabel für ein 10 Mbit/s Basisbandsignal für Entfernungen bis zu 4 km in einer physischen Stern-Topologie (Es gibt drei Unterspezifikationen: 10Base-FL für Glasfaser-Link, 10Base-FB für Glasfaser-Backbone und 10Base-FP für Glasfaser-passiv).
10Base-T	verwendet ein 24 AWG UTP oder S-UTP (Verdrilltes Adernpaar) für ein 10 Mbit/s Basisbandsignal für Entfernungen bis zu 100 m in einer physischen Stern-Topologie.
10Broad36	verwendet ein 75 Ohm Koaxialkabel für 10 Mbit/s ein Breitbandsignal für Entfernungen bis zu 1800 m (oder 3600 m mit Doppelkabeln) in einer physischen Bus-Topologie.
100BaseTX	spezifiziert die 100Mbit/s Übertragung auf 2 Aderpaaren über eine, mit Komponenten der Kategorie 5 realisierte Verkabelung. Kabel, RJ-45-Wanddosen, Patchpanel usw. müssen gemäß dieser Kategorie für eine Übertragungsfrequenz von mindestens 100MHz ausgelegt sein.

Darüber hinaus gibt es noch weitere Übertragungsstandards, wie z. B.: 100Base-T4 (Fast-ETHERNET über verdrillte Adernpaare), 100Base-FX (Fast-ETHERNET über Lichtwellenleiter) oder P802.11 (Wireless LAN) für eine drahtlose Übertragung.

10Base-T, 100BaseTX

Für den WAGO-ETHERNET-Feldbusknoten kann entweder der 10Base-T-Standard oder 100BaseTX genutzt werden.
Der Netzwerkaufbau ist deshalb sehr einfach und günstig mit S-UTP-Kabel als Übertragungsmedium oder mit Leitungen des Typs STP realisiert werden.
Beide Kabelarten sind in jedem Computerhandel erhältlich.

S-UTP-Kabel (Screened-Unshielded Twisted Pair) sind einmalgeschirmte Kabel der Kategorie 5 mit einer Gesamtabschirmung um alle verdrillten ungeschirmten Adernpaare und einer Impedanz von 100 Ohm.

STP-Kabel (Shielded Twisted Pair) sind symmetrische Kabel der Kategorie 5 mit verseilten und einzeln geschirmten Adernpaaren, ein Gesamtschirm ist nicht vorhanden.

Verkabelung der Feldbusknoten

Für den direkten Anschluss eines Feldbusknoten an die Netzwerkkarte des PC benötigen Sie gegebenenfalls ein sogenanntes Cross-Over-Kabel.

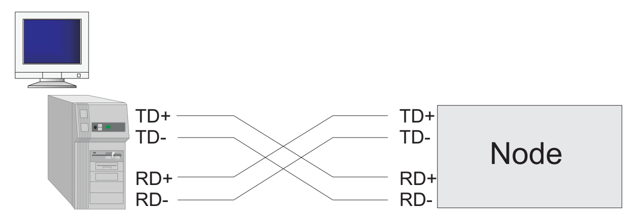


Abbildung 76: Direkter Anschluss eines Knoten mit Cross-Over-Kabel

Zum Verbinden mehrerer Feldbusknoten mit einer Netzwerkkarte, kann der Anschluss der Feldbusknoten über einen sogenannten ETHERNET-Hub mit parallelen Kabeln erfolgen.

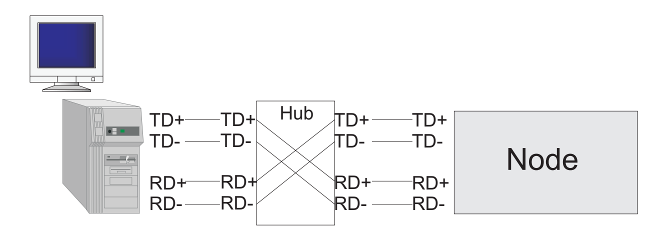
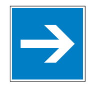


Abbildung 77: Anschluss eines Knoten über ein Hub mit parallelen Kabeln

Ein Hub ist ein Vermittlungssystem zwischen Netzwerksegmenten und Feldbusknoten. Vergleichbar mit einer Mehrfachsteckdose, bildet der Hub den Konzentrationspunkt für eine sternförmige Verkabelung und ermöglicht die Bildung logischer Netzwerke.

Hinweis



Maximale Kabellänge beachten!
Beachten Sie, dass die Kabellänge zwischen Feldbus-Clients und Hub ohne Zwischenschalten von Signalaufbereitungssystemen (z.B. Repeater) maximal 100 m betragen darf. Für größere Netzwerkausdehnungen sind in dem ETHERNET-Standard verschiedene Möglichkeiten beschrieben.



12.1.1.2 Netzwerk-Topologie

Mit 10Base-T, bzw. 100BaseTX werden laut ETHERNET-Standard mehrere Stationen (Knoten) sternförmig verkabelt.
Aus diesem Grund sollen hier lediglich die Stern-Topologie und für größere Netzwerke der Aufbau einer Baum-Topologie genauer betrachtet werden.

Stern-Topologie

Bei der Stern-Topologie handelt es sich um ein Netz, an dem alle Stationen mit einem zentralen Knoten verbunden sind. Dazu wird ein Hub wie ein normaler Rechner an eine Bus-Architektur angeschlossen bzw. verläuft der Bus innerhalb des Hubs.

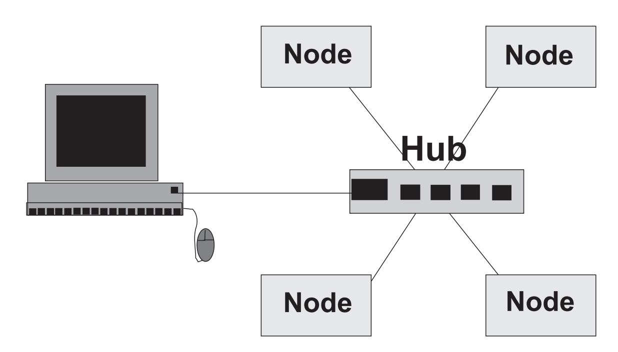


Abbildung 78: Stern-Topologie

Neben der einfachen Realisierung liegen die Vorteile einer solchen Anwendung in der Erweiterbarkeit eines vorhandenen Netzes. Es kann ohne einen Ausfall des Netzes eine Station zugefügt bzw. entnommen werden. Weiterhin wird bei einer defekten Leitung ausschließlich die Kommunikation zur betreffenden Station beeinträchtigt und somit die Ausfallsicherheit des gesamten Netzes deutlich erhöht.

Mit der Stern-Topologie können sehr leicht administrativ zusammengehörende Gruppen gebildet, in hierarchischen Ebenen zusammengefasst und baumartig vernetzt werden.

Baum-Topologie

Bei der Baum-Topologie handelt es sich um eine Struktur, die für größere Netzwerke, z.B. Unternehmen oder Gebäude, eingesetzt wird. Dabei werden verschiedene kleinere Netzwerke beispielsweise über Router hierarchisch wie ein Baum (Äste, Zweige und Stamm) miteinander verbunden.

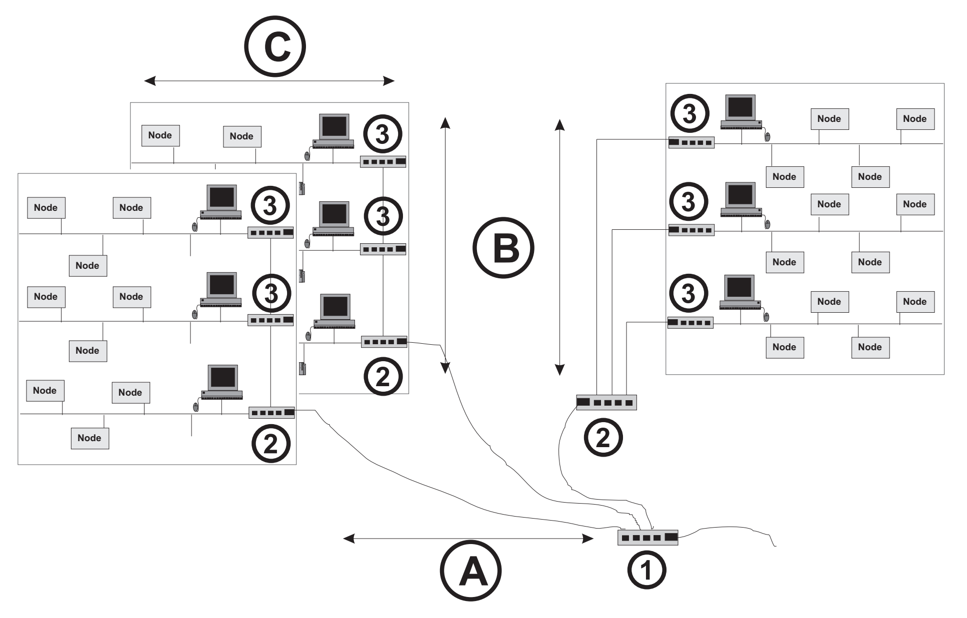


Abbildung 79: Baum-Topologie

Tabelle 66: Legende Baum-Topologie

Beschriftung	Erläuterung
(A)	Primärer Bereich
(B)	Sekundärer Bereich
(C)	Tertiärer Bereich
(1)	Geländeverteiler
(2)	Gebäudeverteiler
(3)	Etagenverteiler

Ein baumartiges Netzwerk wird in drei verschiedene Bereiche aufgeteilt:

Tertiärverkabelung:

Bei der Tertiärverkabelung handelt es sich z. B. um die Vernetzung einer Etage. Das Datenaufkommen ist in diesem Bereich am geringsten.

Sekundärverkabelung:

Die Sekundärverkabelung ist beispielsweise die Verbindung der einzelnen Etagen eines Gebäudes miteinander. Im Vergleich zur Primärverkabelung ist das Datenaufkommen hier deutlich geringer.

Primärverkabelung:

Auf dieser Ebene werden dann die verschiedenen Gebäude miteinander verbunden. Gebäudeübergreifend wird die Verkabelung mittels Lichtwellenleiter empfohlen. Zur Verarbeitung der hohen Datenmengen müssen in diesem Bereich Technologien mit hoher Bandbreite, z. B. durch „Switched ETHERNET“, eingesetzt werden.



Verkabelungsrichtlinien

Allgemeine Richtlinien für den Netzwerkaufbau eines LAN gibt die „Strukturierte Verkabelung„ vor. Darin sind maximal zulässige Kabellängen für die Gelände-, Gebäude- und Etagenverkabelung festgelegt.
Die „Strukturierte Verkabelung“ ist in den Standards EN 50173, ISO 11801 und TIA 568-A normiert. Sie bildet die Grundlage für eine zukunftsweisende, anwendungsunabhängige und wirtschaftliche Netzwerk-Infrastruktur.
Die Verkabelungsstandards definieren einen Geltungsbereich mit einer geographischen Ausdehnung von bis zu 3 km und für eine Bürofläche von bis zu 1 Mio. Quadratmetern mit 50 bis 50.000 Endgeräten. Darüber beschreiben sie Empfehlungen für den Aufbau eines Verkabelungssystems.
Abhängig von der gewählten Topologie, den unter Industriebedingungen eingesetzten Übertragungsmedien und Koppelmodulen sowie von dem Einsatz von Komponenten verschiedener Hersteller in einem Netz, können sich abweichende Angaben ergeben. Die Angaben verstehen sich hier deshalb lediglich als Empfehlungen.

12.1.1.3 Koppelmodule

Es gibt ein Reihe von Koppelmodulen, die bei dem Aufbau eines ETHERNET-Netzwerks eine flexible Gestaltung ermöglichen. Zudem verfügen sie über wichtige Funktionalitäten, die teilweise sehr ähnlich sind.
Deshalb soll die richtige Wahl und die angemessene Verwendung der Module durch eine tabellarische Gegenüberstellung vereinfacht werden.

Tabelle 67: Gegenüberstellung der Koppelmodule für Netzwerke

Modul	Eigenschaft/Verwendung	ISO/OSI-Schicht
Rerpeater	Verstärker zur Signalauffrischung, Verbindung auf physikalischer Ebene.	1
Bridge	Segmentierung von Netzen um die Längenausdehnung zu erweitern.	2
Switch	Multiport-Bridge, d. h. jeder Port verfügt über eine separate Bridge-Funktion. Trennt Netzwerksegmente logisch und verringert dadurch die Netzbelastung. Macht ETHERNET bei konsequentem Einsatz kollisionsfrei.	2 (3)
Hub	Dient dem Aufbau von sternförmigen Topologien, unterstützt unterschiedliche Übertragungsmedien, verhindert keine Netzkollisionen.	2
Router	Verbindung zweier oder mehrerer Datennetze. Angleich von Topologiewechseln und inkompatibler Paketgrößen (z.B. industrieller Bereich und Office-Bereich).	3
Gateway	Verbindung zweier herstellerspezifischer Netze mit unterschiedlicher Soft- und Hardware (z. B. ETHERNET und Interbus-Loop).	4-7

12.1.1.4 ETHERNET-Übertragungsmodus

Einige WAGO-Feldbuskoppler/-controller, die auf ETHERNET basieren, unterstützen sowohl 10Mbit/s als auch 100Mbit/s Übertragungsrate im Voll- bzw. Halbduplex Betrieb.
Um eine sichere und schnelle Übertragung zu gewährleisten, müssen diese Feldbuskoppler/-controller und dessen Link-Partner auf den gleichen Übertragungsmodus konfiguriert sein.

Hinweis



Auf angepassten ETHERNET-Übertragungsmodus achten!
Achten Sie bei der Konfiguration darauf, dass die Übertragungsmodi der Link-Partner übereinstimmen.Eine fehlerhafte Konfiguration des Übertragungsmodus kann einen Linkverlust, eine schlechte Netzwerk-Performance oder ein fehlerhaftes Verhalten des Feldbuskopplers/-controllers zur Folge haben.

Der ETHERNET-Standard IEEE 802.3u sieht zwei Möglichkeiten zur Konfiguration der Übertragungsmodi vor:

- Statische Konfiguration
- Dynamische Konfiguration

Statische Konfiguration der Übertragungsart

Bei der statischen Konfiguration werden beide Link-Partner auf eine statische Übertragungsrate und Duplex-Modus eingestellt. Dabei sind folgende Konfigurationen möglich:

- 10 Mbit/s, Halbduplex
- 10 Mbit/s, Vollduplex
- 100 Mbit/s, Halbduplex
- 100 Mbit/s, Vollduplex

Dynamische Konfiguration der Übertragungsart

Die zweite Konfigurationsmöglichkeit ist der im Standard IEEE 802.3u verankerte Autonegotiation-Modus, bei dem die Übertragungsrate sowie der Duplex-Modus zwischen beiden Kommunikationspartnern dynamisch ausgehandelt werden. Hierbei wird immer der Übertragungsmodus mit der bestmöglichen Performance ermittelt und automatisch von jedem Gerät übernommen.

Hinweis



Autonegation aktivieren!



Achten Sie für ein einwandfreies Funktionieren der dynamischen Konfiguration darauf, dass bei beiden Kommunikationspartnern die Betriebsart der Autonegotiation unterstützt wird und auch aktiviert ist.



Konfigurationsfehler bei der Übertragungsart

Die folgende Liste zeigt eine Aufstellung unzulässiger Konfigurationen:

Tabelle 68: Konfigurationsfehler bei der Übertragungsart

Problem	Ursache	Symptome
Fehlanpassung der Übertragungsrate	Tritt auf, wenn ein Link-Partner mit 10 Mbit/s und der andere mit 100 Mbit/s konfiguriert wurde.	Linkausfall
Fehlanpassung des Duplex-Modus	Tritt auf, wenn ein Link-Partner im Vollduplex- und der andere im Halbduplex-Betrieb arbeitet	Fehlerhafte oder verworfene Datenpakete sowie Kollisionen auf dem Medium
Fehlanpassung bei Autonegotiation	Tritt auf, wenn ein Link-Partner im Autonegotiation-Modus arbeitet und der andere einen statische Konfiguration des Übertragungsmodus im Vollduplexbetrieb verwendet	Der Link-Partner, welcher sich im Autonegotiation-Modus befindet, ermittelt die Netzwerk-Geschwindigkeit über das Parallel-Detection-Verfahren und stellt seinen Duplex-Modus fest auf Halbduplex. Falls das Gerät mit der statischen Konfiguration im Vollduplexbetrieb arbeitet, tritt hier eine Fehlanpassung des Duplex-Modus auf (siehe oben).

12.1.1.5 Wichtige Begriffe

Datensicherheit

Soll ein internes Netz (Intranet) an das öffentliche Netz (z. B. Internet) angeschlossen werden, so ist die Sicherheit der Daten ein sehr wichtiger Aspekt. Durch eine sogenannte Firewall können unerwünschte Zugriffe ausgeschlossen werden.

Bei der Firewall handelt es sich um eine Software oder eine Netzwerkkomponente, die ähnlich einem Router als Koppelglied zwischen Intranet und öffentlichem Netzwerk geschaltet wird. Die Firewall ist in der Lage, Zugriffe ins jeweils andere Netz zu begrenzen oder auch komplett zu sperren, abhängig von der Zugriffsrichtung, dem benutzten Dienst sowie der Identifikation des Netzteilnehmers.

Echtzeitfähigkeit

Oberhalb der Feldbus-Systemebene sind i. Allg. relativ große Datenmengen zu übertragen. Die zulässigen Verzögerungszeiten dürfen ebenfalls verhältnismäßig große Werte annehmen (0,1...10 Sekunden).

Für das Industrie-ETHERNET innerhalb der Feldbus-Systemebene wird hingegen ein Echtzeitverhalten gefordert. Bei ETHERNET kann, z.B. durch die Einschränkung der Busbelastung (< 10 %) oder durch ein Master-Slave-Prinzip, die Erfüllung der Echtzeitanforderungen nahezu realisiert werden.

Das MODBUS/TCP zum Beispiel ist ein sogenanntes Master/Slave-Protokoll. Hierbei sprechen die Slaves lediglich auf die Masterbefehle an. Bei der Verwendung von genau einem Master ergibt sich ein kontrollierter Datenverkehr auf dem Netz und Kollisionen werden vermieden.
Darüber hinaus kann der gezielten Einsatz der Switchingtechnologie die Echtzeitfähigkeit erhöhen.

Shared ETHERNET

Mehrere Knoten, die über einen Hub vernetzt sind, teilen sich ein gemeinsames Medium. Wird von einer Station eine Nachricht gesendet, so wird diese im gesamten Netz veröffentlicht und steht jedem angeschlossenen Knoten zur Verfügung. Die Weiterverarbeitung der Nachricht erfolgt jeweils nur durch den Knoten mit der richtigen Zieladresse. Durch das hohe Datenaufkommen können Kollisionen auftreten und Nachrichten müssen wiederholt übertragen werden. Die Verzögerungszeit ist bei einem Shared ETHERNET so ohne weiteres weder errechenbar noch voraussagbar.

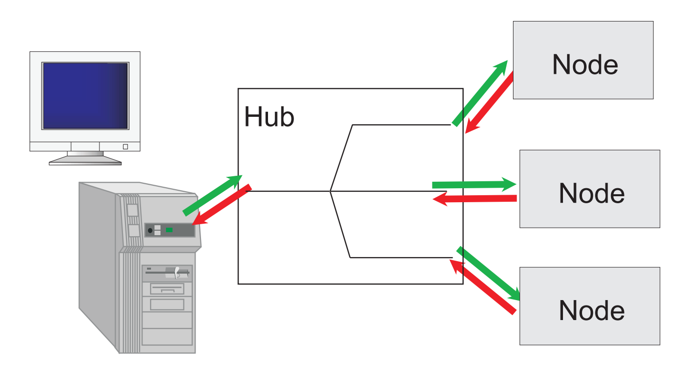


Abbildung 80: Prinzip von Shared ETHERNET

Predictable ETHERNET

Übertragbare Nachrichten können durch die TCP/IP-Software oder das Anwenderprogramm in jedem Teilnehmer begrenzt werden, so dass Echtzeitanforderungen nahezu realisiert werden können. Dabei werden die maximale mittlere Nachrichtenrate (Telegramme pro Sekunde), die maximale mittlere Dauer einer Nachricht und der minimale Zeitabstand zwischen den Nachrichten (Wartezeit des Teilnehmers) beschränkt.
Die Verzögerungszeit einer Nachricht ist damit voraussagbar (predictable).

Switched ETHERNET

Bei einem Switched ETHERNET wird zur Kopplung mehrerer Feldbusknoten ein Switch eingesetzt. Gelangen zu dem Switch Daten aus einem Netzwerksegment, so speichert er diese und prüft, in welches Segment und zu welchem Knoten diese Daten gesendet werden sollen.

Die Nachricht wird dann ausschließlich an den Knoten mit der richtigen Zieladresse übermittelt. Das Datenaufkommen im Netz wird verringert, die Bandbreite erhöht und Kollisionen verhindert. Die Laufzeiten können definiert und berechnet werden, das Switched ETHERNET ist deterministisch.



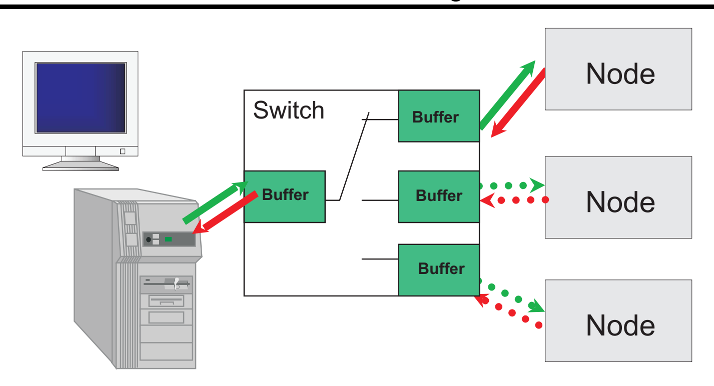


Abbildung 81: Prinzip von Switched ETHERNET

12.1.2 Netzwerkkommunikation

12.1.2.1 ETHERNET-Datenpaket

Die auf dem Übertragungsmedium ausgetauschten Telegramme werden „Paket“ oder „ETHERNET-Paket“ genannt. Die Übertragung erfolgt verbindungslos, d. h. der Sender erhält keine Rückmeldung von dem Empfänger. Die Nutzdaten werden in einen Rahmen von Adressinformationen gepackt. Der Aufbau eines solchen Paketes ist in der folgenden Abbildung dargestellt.

Tabelle 69: ETHERNET-Datenpaket

Präamble	ETHERNET-Header	ETHERNET-Nutzdatenbereich	Prüfsumme
8 Byte	14 Byte	46-1500 Byte	4 Byte

Die Präamble dient zur Synchronisation zwischen Sende- und Empfangsstation.

Der ETHERNET-Header beinhaltet die MAC-Adressen des Senders und des Empfängers und ein Typfeld. Das Typfeld dient zur Identifikation des nachfolgenden Protokolls mittels einer eindeutigen Kodierung (z. B. 0800hex = Internet Protokoll).

12.1.2.2 Adressierung (MAC-ID)

Jeder ETHERNET basierende Feldbuskoppler/-controller von WAGO erhält bereits bei seiner Fabrikation eine einmalige und weltweit eindeutige physikalische ETHERNET-Adresse, auch MAC-ID (Media Access Control Identity) genannt.

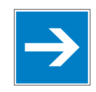
Diese kann von dem Netzwerkbetriebssystem zur Adressierung auf Hardware-Ebene verwendet werden.

Die Adresse besitzt eine feste Länge von 6 Byte (48 Bit) und beinhaltet den Adresstyp, die Kennzeichnung für den Hersteller und die Seriennummer.

Beispiel für die MAC-ID eines WAGO-ETHERNET TCP/IP-Controller (hexadezimal): 00H-30H-DEH-00H-00H-01H.

Die Adressierung verschiedener Netze ist mit ETHERNET nicht möglich.
Soll ein ETHERNET-Netzwerk mit anderen Netzen verbunden werden, muss deshalb mit übergeordneten Protokollen gearbeitet werden.

Hinweis



Netzverbindungen über Router herstellen!
Um zwei oder mehr Datennetze miteinander zu verbinden, setzen Sie Router ein.



12.1.2.3 Buszugriffsverfahren CSMA/CD

Der Zugriff der Feldbusknoten auf den Bus geschieht beim ETHERNET-Standard über das sogenannte Konkurrenzverfahren CSMA/CD (Carrier Sense Multiple Access/ Collision Detection).

- Carrier Sense: Der Sender horcht auf dem Bus.
- Multiple Access: Mehrere Sender können auf den Bus zugreifen.
- Collision Detection: Eine „Kollision“ wird erkannt.

Dabei kann jede Station eine Nachricht senden, nachdem sie sich davon überzeugt hat, dass das Übertragungsmedium frei ist. Treten Kollisionen von Datenpaketen durch zeitgleiches Senden mehrerer Stationen auf, sorgt CSMA/CD dafür, dass solche erkannt und die Datenübertragung wiederholt wird.

Für eine sichere Datenübertragung unter Industriebedingungen reicht das jedoch nicht aus. Damit die Kommunikation und Datenübertragung über ETHERNET zuverlässig erfolgen kann, werden verschiedene Kommunikationsprotokolle benötigt.

12.1.3 Protokoll-Schichtenmodell (Beispiel)

In der Darstellung des Schichtenmodells soll anhand eines Beispiels (MODBUS-Protokoll und EtherNet/IP) die Einordnung und die Zusammenhänge der in einem ETHERNET basierenden WAGO-Feldbuskoppler/-controller verdeutlicht werden.

In diesem Beispiel kann die Kommunikation entweder über das MODBUS-Protokoll oder über EtherNet/IP erfolgen.

1 ETHERNET:
Als Grundlage für den physikalischen Datenaustausch dient die ETHERNET-Hardware. Die auszutauschenden Datensignale und das Buszugriffsverfahren CSMA/CD sind in einem Standard festgelegt.

Tabelle 70: Physical Layer

	...
(1)	ETHERNET (physikalisches Interface, CSMA/CD)

2 IP:
Für die Kommunikation ist über der ETHERNET-Hardware das Internet Protocol (IP) angeordnet. Dieses bündelt die zu übertragenden Daten in Pakete mit Absender- und Empfängeradresse und gibt diese Pakete nach unten an die ETHERNET-Schicht zur physikalischen Übertragung weiter. Auf der Empfängerseite nimmt IP die Pakete von der ETHERNET-Schicht in Empfang und packt sie aus.

Tabelle 71: Network Layer

	...
(2)	IP
(1)	ETHERNET (physikalisches Interface, CSMA/CD)

3 TCP/UDP:

- TCP: (Transmission Control Protocol)
  Das über der IP-Schicht angeordnete TCP-Protokoll überwacht den Transport der Datenpakete, sortiert deren Reihenfolge und fordert fehlende Pakete erneut an. TCP ist ein verbindungsorientiertes Transport-Protokoll. Zusammengefasst werden die Protokoll-Schichten TCP und IP auch TCP/IP-Protokoll-Stapel oder TCP/IP-Stack genannt.
- UDP: (User Datagram Protocol)
  Die UDP-Schicht ist ebenso wie TCP ein Transport Protokoll, welches über der IP-Schicht angeordnet ist. Im Vergleich zum TCP-Protokoll ist UDP nicht verbindungsorientiert. Das heißt es gibt keine Kontrollmechanismen für den Datenaustausch zwischen Sender und Empfänger.
  Der Vorteil dieses Protokolls liegt in der Effizienz der übertragenen Daten und damit in der resultierenden höheren Verarbeitungsgeschwindigkeit.



Viele Programme nutzen beide Protokolle. Wichtige Status-Informationen werden über die zuverlässige TCP-Verbindung gesendet, während der Hauptstrom der Daten über UDP versendet wird.

Tabelle 72: Transport Layer

	...
(3)	TCP/UDP
(2)	IP
(1)	ETHERNET (physikalisches Interface, CSMA/CD)

4 Konfigurations-, Diagnose und Anwendungsprotokolle:
Auf dem TCP/IP-Stack oder auf der UDP/IP-Schicht setzen entsprechend implementierte Konfigurations-, Diagnose- und Anwenderprotokolle auf, die anwendergerechte Dienste zur Verfügung stellen. Zur Konfiguration und Diagnose sind dieses z. B. SMTP (Simple Mail Transport Protocol) für Emails, HTTP (Hypertext Transport Protokoll) für WWW-Browser und einige andere.

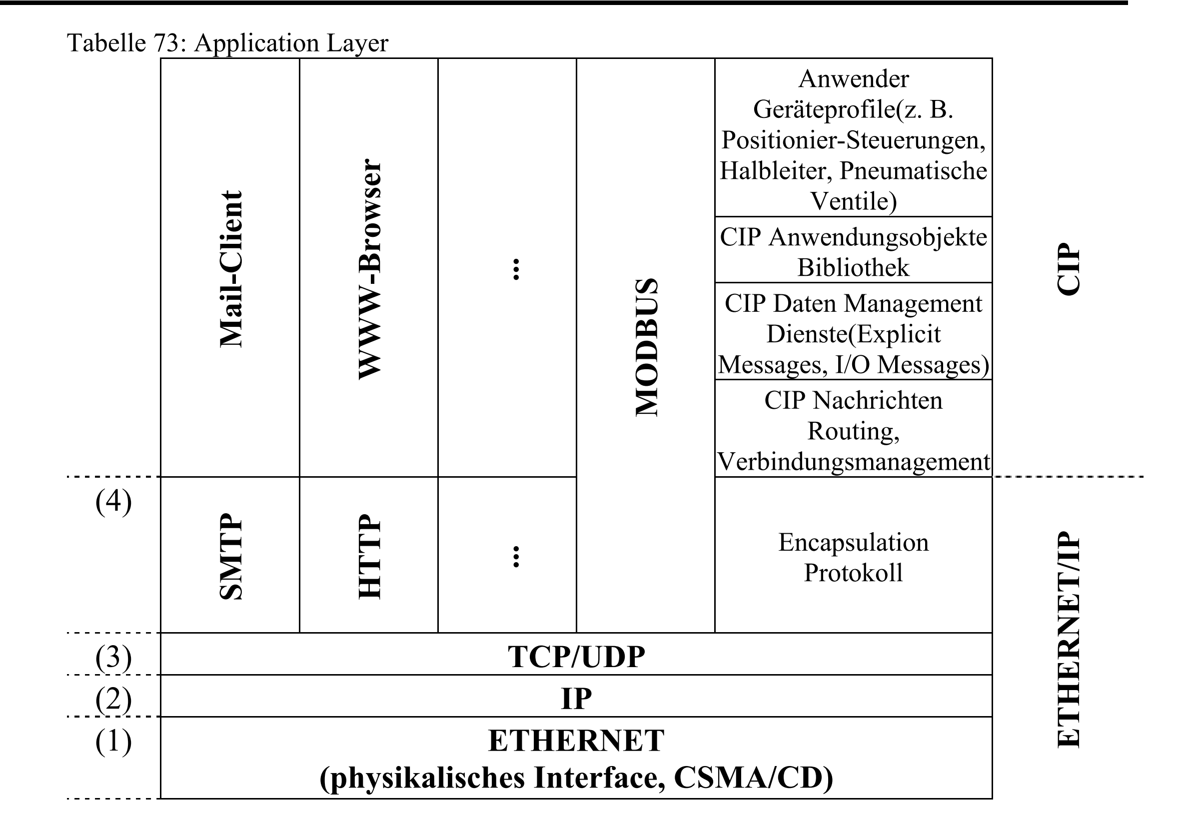
Für die Anwendung in der industriellen Datenkommunikation sind in dieser Darstellung als Beispiel die Protokolle MODBUS/TCP (UDP) und EtherNet/IP implementiert.

Das MODBUS-Protokoll setzt dabei ebenfalls direkt auf TCP (UDP)/IP auf,

EtherNet/IP hingegen besteht zusammengefasst aus den Protokollschichten ETHERNET, TCP und IP mit einem darauf aufsetzenden Encapsulation Protokoll. Dieses dient zur Anbindung an CIP (Control and Information Protocol).

CIP wird in gleicher Weise, wie von EtherNet/IP, auch von DeviceNet verwendet. Dadurch lassen sich Applikationen mit DeviceNet-Geräteprofilen sehr einfach auf EtherNet/IP überführen.

Tabelle 73: Application Layer



	Mail-Client	WWW-Browser	...	MODBUS	Anwender Geräteprofile(z. B. Positionier-Steuerungen, Halbleiter, Pneumatische Ventile)	CIP
					CIP Anwendungsobjekte Bibliothek	
					CIP Daten Management Dienste(Explicit Messages, I/O Messages)	
					CIP Nachrichten Routing, Verbindungsmanagement	
(4)	SMTP	HTTP	...		Encapsulation Protokoll	ETHERNET/IP
(3)	**TCP/UDP**					
(2)	**IP**					
(1)	**ETHERNET (physikalisches Interface, CSMA/CD)**					

Datenstruktur

Die nachfolgende Darstellung zeigt den Aufbau der Datenstrukturen, die aus diesen Protokollen resultieren. Dabei wird deutlich, wie die Datenpakete der Kommunikationsprotokolle ETHERNET, TCP und IP mit dem aufsetzenden Anwendungsprotokoll MODBUS für eine Übertragung ineinander verschachtelt werden.

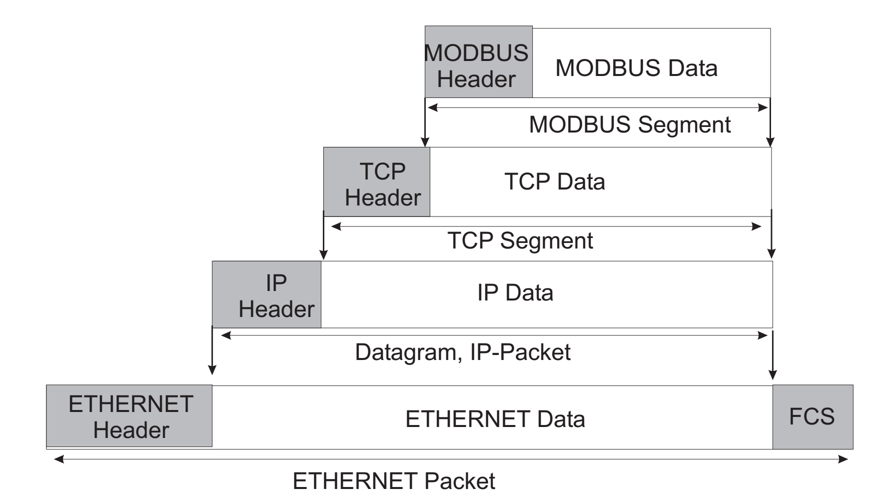


Abbildung 82: Aufbau der Datenstrukturen bei verschachtelten Protokollen



12.1.4 Kommunikationsprotokolle

12.1.4.1 IP (Internet Protocol)

Das Internet Protokoll (IP) teilt Datentelegramme in Segmente und ist verantwortlich für deren Beförderung von einem Netzteilnehmer zu einem anderen. Die beteiligten Stationen können sich dabei in dem selben Netzwerk befinden oder in verschiedenen physikalischen Netzwerken, die aber mit Routern miteinander verbunden sind.

Die Router sind in der Lage, verschiedene Pfade (Netzwerkübertragungswege) durch einen Netzwerkverbund auszuwählen und somit Überlastungen und Störungen einzelner Netze zu umgehen.

Dabei kann es jedoch vorkommen, dass einzelne Strecken gewählt werden, die kürzer sind als andere. Daraufhin können sich Telegramme überholen und die Reihenfolge (Sequenz) der Datenpakete ist falsch.

Die Gewährleistung der korrekten Übertragung muss deshalb in höheren Schichten, z. B. durch TCP erfolgen.

IP-Datenpaket

Die IP-Datenpakete enthalten neben den zu transportierenden Nutzdaten eine Fülle von Adress- und Zusatzinformationen in dem „Paketkopf".

Tabelle 74: IP-Datenpaket

IP-Header	IP-Nutzdatenbereich

Die wichtigsten Informationen in dem IP-Header sind die IP-Adressen vom Absender und Empfänger sowie das benutzte Transportprotokoll.

IP-Adressen

Für die Kommunikation im Netz muss jeder Feldbusknoten über eine 32-Bit lange Internet-Adresse (IP Adresse) verfügen.

Hinweis



IP-Adressen müssen einmalig sein!
Zum fehlerfreien Betrieb muss die eingestellte IP-Adresse im gesamten Netzwerkverbund einmalig sein.

Wie unten aufgezeigt, gibt es verschiedene Adressklassen mit unterschiedlich langer Netzwerk-Identifikation (Net-ID) und Host-Rechner-Identifikation (Host-ID).
Die Net-ID definiert das Netzwerk, in dem sich der Teilnehmer befindet.
Die Host-ID identifiziert einen bestimmten Teilnehmer innerhalb dieses Netzwerkes.

Zur Adressierung werden Netze in mehrere Netzwerkklassen unterteilt:

- **Class A:** (Net-ID: Byte 1, Host-ID: Byte 2… Byte 4)

Tabelle 75: Netzwerkklasse Class A

<table>
<tr><td>z. B.</td><td>101 .</td><td>16 .</td><td>232 .</td><td>22</td></tr>
<tr><td colspan="2">01100101</td><td>00010000</td><td>11101000</td><td>00010110</td></tr>
<tr><td>0</td><td>Net-ID</td><td colspan="3">Host-ID</td></tr>
</table>

Das höchste Bit bei Class A-Netzen ist immer '0'.
D. h., das höchste Byte kann im Bereich von '0 0000000' bis '0 1111111' liegen.
Der Adressbereich der Class A-Netze liegt somit im ersten Byte immer zwischen 0 und 127.

- **Class B:** (Net-ID: Byte 1 … Byte 2, Host-ID: Byte 3… Byte 4)

Tabelle 76: Netzwerkklasse Class B

<table>
<tr><td>z. B.</td><td>181 .</td><td>16 .</td><td>232 .</td><td>22</td></tr>
<tr><td colspan="2">10110101</td><td>00010000</td><td>11101000</td><td>00010110</td></tr>
<tr><td>10</td><td colspan="2">Net-ID</td><td colspan="2">Host-ID</td></tr>
</table>

Die höchsten Bits bei Class B-Netzen sind immer '10'.
D. h., das höchste Byte kann im Bereich von '10 000000' bis '10 111111' liegen.
Der Adressbereich der Class B-Netze liegt somit im ersten Byte immer zwischen 128 und 191.

- **Class C:** (Net-ID: Byte 1 … Byte 3, Host-ID: Byte 4)

Tabelle 77: Netzwerkklasse Class C

<table>
<tr><td>z. B.</td><td>201 .</td><td>16 .</td><td>232 .</td><td>22</td></tr>
<tr><td colspan="2">11000101</td><td>00010000</td><td>11101000</td><td>00010110</td></tr>
<tr><td>110</td><td colspan="3">Net-ID</td><td>Host-ID</td></tr>
</table>

Die höchsten Bits bei Class C-Netzen sind immer '110'.
D. h., das höchste Byte kann im Bereich von '110 00000' bis '110 11111' liegen.
Der Adressbereich der Class C-Netze liegt somit im ersten Byte immer zwischen 192 und 223.

- **Weitere Netzwerkklassen (D, E):** werden für Sonderaufgaben verwendet.



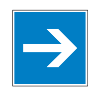
Eckdaten

Tabelle 78: Eckdaten Class A, B und C

Netzwerkklasse	Adressbereich des Netzwerkteils	Mögliche Anzahl von	
		Netzen	Hosts pro Netz
Class A	1.XXX.XXX.XXX ... 126.XXX.XXX.XXX	127 (2^7)	Ca. 16 Millionen (2^{24})
Class B	128.000.XXX.XXX ... 191.255.XXX.XXX	Ca. 16 Tausend (2^{14})	Ca. 65 Tausend (2^{16})
Class C	192.000.000.XXX ... 223.255.255.XXX	Ca. 2 Millionen (2^{21})	254 (2^8)

Jedem ETHERNET basierenden Koppler oder Controller kann über das implementierte BootP-Protokoll sehr leicht eine IP Adresse zugeteilt werden. Als Empfehlung für ein kleines internes Netzwerk gilt hier Netzwerk-Adressen aus dem Class C-Bereich zu wählen.

Hinweis



Bei IP-Adressen nicht 0 und 255 verwenden!
Beachten Sie, dass niemals alle Bits in einem Byte gleich ‚0' oder gleich ‚1' gesetzt sind (Byte = 0 oder 255). Diese sind für spezielle Funktionen reserviert und dürfen nicht vergeben werden. So darf z. B. darf die Adresse 10.0.10.10 wegen der 0 im zweiten Byte nicht verwendet werden.

Soll ein Netzwerk direkt mit dem Internet verbunden werden, so werden von einer zentralen Vergabestelle zugeteilte weltweit einmalige IP-Adressen verwendet. Die Vergabe in Deutschland erfolgt z. B. durch die DENIC eG (Deutsches Network Information Center) in Karlsruhe.

Hinweis



Internetanbindung nur durch autorisierten Netzwerkadministrator!
Beachten Sie, dass eine direkte Internetanbindung ausschließlich durch einen autorisierten Netzwerkadministrator erfolgen darf, deshalb ist eine solche Anbindung nicht in diesem Handbuch beschrieben.

Subnetzwerke

Um das Routing innerhalb von großen Netzwerken zu ermöglichen, wurde in der Spezifikation RFC 950 eine Konvention eingeführt. Dabei wird ein Teil der Internet-Adresse, die Host-ID, weiter unterteilt und zwar in eine Subnetzwerknummer und die eigentliche Stationsnummer des Knoten. Mit Hilfe der Netzwerknummer kann nun innerhalb des Teilnetzwerkes in interne Unternetzwerke verzweigt werden, von außen aber ist das gesamte Netzwerk als Einheit sichtbar. Größe und Lage der Subnetzwerk-ID sind nicht festgeschrieben, die Größe ist jedoch abhängig von der Anzahl der zu adressierenden Subnetze und die Anzahl der Hosts pro Subnetz.

Tabelle 79: Klasse B-Adresse mit Feld für Subnetzwerk-ID

1			8	16	24	32
1	0	...	Netz-ID	Subnetz-ID	Host-ID	



Subnetz-Maske

Für die Kodierung der Subnetze im Internet, wurde die sogenannte Subnetz-Maske eingeführt. Dabei handelt es sich um eine Bit-Maske, mit der spezielle Bits der IP-Adresse ausgeblendet bzw. selektiert werden können. Die Maske definiert, welche Bits der Host-ID für die Subnetz-Kodierung verwendet werden und welche die ID des Hosts bezeichnen.
Der gesamte IP-Adressbereich liegt theoretisch zwischen 0.0.0.0 und 255.255.255.255. Für die Subnetz-Maske sind jeweils die 0 und die 255 aus dem IP-Adressbereich reserviert.

Die von der jeweiligen Netzwerkklasse abhängigen Standard-Masken sehen wie folgt aus:

- **Class A-Subnetz-Maske:**

Tabelle 80: Subnetz-Maske für Class A-Netzwerke

255	.0	.0	.0

- **Class B-Subnetz-Maske:**

Tabelle 81: Subnetz-Maske für Class B-Netzwerke

255	.255	.0	.0

- **Class C-Subnetz-Maske:**

Tabelle 82: Subnetz-Maske für Class C-Netzwerke

255	.255	.255	.0

Je nach Subnetz-Unterteilung, können die Subnetz-Masken über 0 und 255 hinaus aber auch andere Werte enthalten, wie z. B. 255.255.255.128 oder 255.255.255.248, usw.

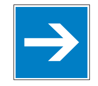
Die Subnetz-Masken-Nummer wird von dem Netzwerkadministrator zugewiesen. Zusammen mit der IP-Adresse bestimmt diese Nummer, zu welchem Netzwerk der PC und der Knoten gehört.

Der Empfängerknoten, der sich in einem Subnetz befindet, berechnet zunächst die richtige Netzwerknummer aus seiner eigenen IP Adresse und der Subnetzwerk-Maske. Erst im Anschluss daran, überprüft er die Knotennummer und liest dann bei Übereinstimmung den gesamten Paket-Rahmen aus.

Tabelle 83: Beispiel für eine IP-Adresse aus einem Class B-Netz

IP-Adresse	172.16.233.200	10101100 00010000 11101001 11001000
Subnetz-Maske	255.255.255.128	11111111 11111111 11111111 10000000
Netz-ID	172.16.0.0	10101100 00010000 00000000 00000000
Subnetz-ID	0.0.233.128	00000000 00000000 11101001 10000000
Host-ID	0.0.0.72	00000000 00000000 00000000 01001000

Hinweis



Angabe der Netzwerk-Maske erforderlich!
Beachten Sie, dass die vom Administrator festgelegte Netzwerk-Maske bei der Installation des Netzwerkprotokolls genauso wie die IP-Adresse angegeben werden muss.



Gateway

Die Subnetze des Internets sind in der Regel über Gateways verbunden. Diese Gateways dienen dazu, Pakete an andere Netzwerke oder Subnetze weiterzuleiten. Für einen an das Internet angeschlossenen PC oder Feldbusknoten bedeutet das, dass zusätzlich zur IP-Adresse und Netzwerk Maske für jede Netzwerkkarte die korrekte IP Adresse des Standard-Gateways angegeben werden muss. Diese IP-Adresse sollte Ihnen ebenfalls von Ihrem Netzwerkadministrator zur Verfügung gestellt werden.
Ohne Angabe dieser Adresse bleibt die IP-Funktionalität auf das lokale Subnetz beschränkt.

RAW-IP

Raw-IP kommt ohne Protokolle, wie z. B. PPP (Punkt-zu-Punkt-Protokoll) aus. Bei RAW-IP werden die TCP/IP-Pakete direkt, ohne Handshaking ausgetauscht, wodurch ein schnellerer Verbindungsaufbau möglich ist. Zuvor muss allerdings die Konfiguration mit einer festen IP-Adresse stattgefunden haben. Vorteile von RAW-IP sind eine hohe Datentransferrate und eine gute Stabilität.

IP-Multicast

Unter Multicast versteht man eine Übertragungsart von einem Punkt zu einer Gruppe, also eine Punkt-zu-Mehrpunkt-Übertragung oder auch Mehrpunktverbindung genannt. Der Vorteil von Multicast liegt darin, dass gleichzeitig Nachrichten über eine Adresse an mehrere Teilnehmer oder geschlossene Teilnehmergruppen (Closed User Groups) übertragen werden.
IP-Multicasting auf der Internetwork-Ebene wird durch das Internet Group Message Protocol IGMP realisiert; dieses Protokoll wird von Nachbar-Routern benutzt, um sich gegenseitig über Gruppenzugehörigkeiten zu informieren.
Bei der Verteilung von Multicast-Paketen im Subnetwork geht IP davon aus, dass der Datalink-Layer seinerseits Multicasting zur Verfügung stellt. Im Falle ETHERNET sind Multicast-Adressen vorhanden, mit denen ein durch sie adressiertes Paket durch eine einzige Sendeoperation an mehrere Empfänger verschickt wird. Hier stützt man sich darauf, dass ein gemeinsames Medium die Möglichkeit bietet, Pakete an mehrere Empfänger gleichzeitig zu senden. Die Stationen untereinander müssen sich nicht informieren, wer zu einer Multicast-Adresse gehört - jede Station empfängt physikalisch jedes Paket.
Die Adressauflösung von IP-Adresse zu ETHERNET-Adresse wird algorithmisch gelöst, IP-Multicast-Adressen werden in ETHERNET-Multicastadressen eingebettet.

12.1.4.2 TCP (Transmission Control Protocol)

Aufgesetzt auf das Internet-Protokoll, übernimmt TCP (Transmission Control Protocol) die Sicherung des Datentransportes durch das Netzwerk.
Dazu stellt TCP für die Dauer der Datenübertragung eine Verbindung zwischen zwei Teilnehmern her. Die Kommunikation erfolgt im Voll-Duplexverfahren, d. h. beide Teilnehmer können gleichzeitig Daten empfangen und versenden.
Die übertragenen Nutzdaten werden von TCP mit einer 16 bit-Prüfsumme versehen und jedes Datenpaket erhält eine Sequenznummer.

Der Empfänger überprüft anhand der Prüfsumme den korrekten Empfang des Paketes und verrechnet anschließend die Sequenznummer. Das Ergebnis nennt sich Acknowledgement-Nr. und wird mit dem nächsten selbst versendeten Paket als Quittung zurückgesendet. Dadurch ist gewährleistet, dass der Verlust von TCP-Paketen bemerkt wird, und diese im Bedarfsfall in korrekter Abfolge erneut gesendet werden können.

TCP-Datenpaket

Der Paketkopf eines TCP-Datenpaketes besteht aus mindestens 20 Byte und enthält unter anderem die Portnummer der Applikation des Absenders sowie die des Empfängers, die Sequenznummer und die Acknowledgement-Nr.
Das so entstandene TCP-Paket wird in den Nutzdatenbereich eines IP-Paketes eingesetzt, so dass ein TCP/IP-Paket entsteht.

TCP-Portnummern

TCP kann zusätzlich zur IP-Adresse (Netz- und Host-Adresse) gezielt eine spezielle Anwendung (Dienst) auf dem adressierten Host ansprechen. Dazu werden die auf einem Host befindlichen Anwendungen, wie z. B. Web-Server, FTP-Server und andere, über unterschiedliche Portnummern adressiert. Für bekannte Anwendungen werden feste Ports vergeben, auf die sich jede Anwendung beim Verbindungsaufbau beziehen kann.
(Beispiele: Telnet-Portnummer: 23, HTTP-Portnummer: 80).
Eine komplette Liste der „normierten Dienste“ findet sich in den Spezifikationen RFC 1700 (1994).

12.1.4.3 UDP (User Datagram Protocol)

Das UDP-Protokoll ist, wie auch das TCP-Protokoll, für den Datentransport zuständig. Im Vergleich zum TCP-Protokoll ist UDP nicht verbindungsorientiert. Das heißt es gibt keine Kontrollmechanismen bei dem Datenaustausch zwischen Sender und Empfänger. Der Vorteil dieses Protokolls liegt in der Effizienz der übertragenen Daten und damit in der resultierenden höheren Verarbeitungsgeschwindigkeit.

12.1.4.4 ARP (Adress Resolution Protocol)

ARP (Abkürzung für „Address Resolution Protocol“).
Dieses Protokoll verbindet die IP-Adresse mit der physikalischen MAC-Adresse der jeweiligen ETHERNET-Karte. Es kommt immer dann zum Einsatz, wenn die Datenübertragung zu einer IP-Adresse im gleichen logischen Netz erfolgt, in dem sich auch der Absender befindet.

12.1.5 Konfigurations- und Diagnoseprotokolle

12.1.5.1 BootP (Bootstrap Protocol)

Mit dem „Bootstrap Protocol“ (BootP) können Sie dem Feldbuskoppler/-controller in einem TCP/IP-Netzwerk eine IP-Adresse und andere Parameter zuweisen. Außerdem können Subnetzmaske und Gateway übermittelt werden.



Die Protokollkommunikation besteht aus einer Client-Anfrage des Feldbuskopplers und einer Server-Antwort von dem PC.

Über das Protokoll wird eine Broadcast-Anfrage auf Port 67 (BootP-Server) gesendet, welche die Hardware-Adresse (MAC-ID) des Feldbuskopplers enthält.

Der BootP-Server erhält die Nachricht. Er beinhaltet eine Datenbank, in dem MAC-ID und IP-Adressen einander zugeordnet sind. Wird die MAC-Adresse gefunden, wird eine Broadcast-Antwort über das Netz gesendet.

Der Feldbuskoppler/-controller lauscht auf dem vorgegebenen Port 68 auf die Antwort des BootP-Servers. Ankommende Pakete enthalten unter anderem die IP-Adresse und die MAC-Adresse des Feldbuskopplers/-controllers. An der MAC-Adresse erkennt ein Feldbuskoppler/-controller, ob die Nachricht für ihn bestimmt ist und übernimmt bei Übereinstimmung die gesendete IP-Adresse in sein Netzwerk-Interface.

Hinweis



IP-Adressvergabe über BootP unter Windows und Linux möglich!
Sie können eine IP-Adresse mittels WAGO-BootP-Server sowohl unter Windows- als auch unter Linux-Betriebssystemen vergeben.
Neben dem WAGO-BootP-Server können Sie aber auch beliebige andere BootP-Server verwenden.



Information



Weitere Informationen zu dem WAGO-BootP-Server
Sie erhalten den „WAGO-BootP-Server 759-315“ kostenlos auf der DVD-ROM „AUTOMATION Tools and Docs“ (Art.-Nr.: 0888-0412) oder auf der Internetseite http://www.wago.com unter Downloads → AUTOMATION → 759-315 WAGO-BootP-Server.

Information



Weitere Information zur Adressvergabe mit BootP-Server
Die Vorgehensweise der Adressvergabe mit dem WAGO-BootP-Server ist detailliert in dem Kapitel „Feldbusknoten in Betrieb nehmen“ beschrieben.

Der BootP-Client dient zum dynamischen Konfigurieren der Netzwerkparameter:

Tabelle 84: Bedeutung der BootP-Parameter

Parameter	Bedeutung
IP-Adresse des Clients	Netzwerk-Adresse des Feldbuskopplers/-controllers
IP-Adresse des Routers	Falls eine Kommunikation außerhalb des lokalen Netzwerkes stattfinden soll, wird die IP-Adresse des Routers (Gateway) in diesem Parameter angegeben.
Subnetmask	Die Subnetzmaske ermöglicht dem Feldbuskoppler/-controller zu unterscheiden, welche Teile der IP-Adresse das Netzwerk und welche die Netzwerkstationen bestimmen.
IP-Adressen der DNS-Server	Hier können die IP-Adressen von maximal 2 DNS-Servern angegeben werden.
Hostname	Name des Hosts

Bei der Verwendung des Bootstrap Protokolls zur Konfiguration des Knotens, werden die Netzwerkparameter (IP-Adresse, etc...) im EEPROM abgelegt.

Hinweis



BootP-Konfiguration wird im EEPROM gespeichert!
Beachten Sie, dass die Netzwerk-Konfiguration bei der Verwendung von BootP im Gegensatz zu der Konfiguration über DHCP im EEPROM abgelegt wird.

Standardmäßig ist im Feldbuskoppler/-controller das BootP aktiviert.

Bei aktiviertem BootP erwartet der Feldbuskoppler/-controller die permanente Anwesenheit eines BootP-Servers.
Ist jedoch nach einem PowerOn-Reset kein BootP-Server verfügbar, dann bleibt das Netzwerk inaktiv.

Um den Feldbuskoppler/-controller mit der in dem EEPROM hinterlegten IP-Konfiguration zu betreiben, ist BootP nach der Konfiguration zu deaktivieren. Dieses erfolgt z. B. über das Web-based Management-System auf der entsprechenden feldbuskoppler/-controller-internen HTML-Seite, die unter dem Link: „Port" zu erreichen ist.

Ist das BootP deaktiviert, verwendet der Feldbuskoppler/-controller beim nächsten Bootvorgang die im EEPROM abgespeicherten Parameter.

Bei einem Fehler in den abgespeicherten Parametern wird über die I/O-LED ein Blinkcode ausgegeben und die Konfiguration über BootP automatisch eingeschaltet.

12.1.5.2 HTTP (Hypertext Transfer Protocol)

HTTP ist ein Protokoll, das von WWW (World Wide Web)-Servern zur Weitergabe von Hypermedien, Text, Bildern, Audiodaten usw. verwendet wird. Das HTTP bildet heutzutage die Grundlage des Internets und basiert ebenso wie das BootP-Protokoll auf Anforderungen und Antworten.

Der auf dem Feldbuskoppler/-controller implementierte HTTP-Server dient zum Auslesen der im Feldbuskoppler/-controller abgespeicherten HTML-Seiten. Die HTML-Seiten geben Auskunft über den Feldbuskoppler/-controller (Zustand, Konfiguration), das Netzwerk und das Prozessabbild.

Auf einigen HTML-Seiten können auch Feldbuskoppler/-controller-Einstellungen über das Web-based Management-System festgelegt und geändert werden, z. B., ob die Netzwerk-Konfiguration des Feldbuskoppler/-controller über das DHCP, das BootP-Protokoll oder aus den gespeicherten Daten im EEPROM erfolgen soll.

Der HTTP-Server benutzt die Portnummer 80.

12.1.5.3 DHCP (Dynamic Host Configuration Protocol)

Die über den Link: „Port" zu öffnende feldbuskoppler/-controller-interne HTML-Seite bietet die Option, die Netzwerk-Konfiguration anstatt mit dem BootP-



Protokoll auch über die im EEPROM gespeicherten Daten oder über das DHCP durchzuführen.

DHCP (Dynamic Host Configuration Protocol) ist eine Weiterentwicklung von BootP und ist mit diesem rückwärts kompatibel.
Sowohl BOOTP als auch DHCP weisen dem Feldbusknoten (Client) beim Starten eine IP-Adresse zu, der Ablauf ist dabei der gleiche wie bei BootP.

Bei der Konfiguration der Netzwerkparameter über DHCP, sendet der Feldbuskoppler/-controller nach der Initialisierung eigenständig eine Client Anfrage an den DHCP-Server z. B. auf dem angeschlossenen PC.

Über das Protokoll wird eine Broadcast-Anfrage auf Port 67 (DHCP-Server) gesendet, welche die Hardware-Adresse (MAC-ID) des Feldbuskopplers/-controllers enthält.

Der DHCP-Server erhält die Nachricht. Er beinhaltet eine Datenbank, in dem MAC-ID und IP-Adressen einander zugeordnet sind. Wird die MAC-Adresse gefunden, wird eine Broadcast-Antwort über das Netz gesendet.

Der Feldbuskoppler/-controller wartet auf dem vorgegebenen Port 68 auf die Antwort des DHCP-Servers. Ankommende Pakete enthalten unter anderem die IP-Adresse und die MAC-Adresse des Feldbuskopplers/-controllers. An der MAC-Adresse erkennt ein Feldbuskoppler/-controller, ob die Nachricht für ihn bestimmt ist und übernimmt bei Übereinstimmung die gesendete IP-Adresse in sein Netzwerk-Interface.

Erfolgt keine Antwort, so wird die Anfrage nach 4 Sekunden, eine weitere nach 8 Sekunden und nach 16 Sekunden gesendet.
Bleiben alle Anfragen ohne Antwort, so wird ein Blinkcode über die I/O-LED ausgegeben. Eine Übernahme der Parameter aus dem EEPROM ist nicht möglich.

Hinweis



DHCP-Konfiguration wird nicht im EEPROM gespeichert!
Beachten Sie, dass die Netzwerk-Konfiguration über DHCP im Gegensatz zu der Verwendung von BootP nicht im EEPROM abgelegt wird.

Der Unterschied zwischen BOOTP und DHCP besteht darin, dass beide verschiedene Zuordnungsverfahren verwenden.
BOOTP ermöglicht die Zuordnung einer festen IP-Adresse für jeden Client, wobei diese Adressen und ihre Reservierung ständig in der BOOTP-Serverdatenbank gespeichert sind.
DHCP ermöglicht dagegen zusätzlich die dynamische Zuordnung verfügbarer IP-Adressen durch Clientleases (Lease-Time, nach der der Client eine neue Adresse anfragt), wobei jede DHCP-Clientadresse temporär in der Serverdatenbank gespeichert ist.

Darüber hinaus ist für DHCP-Clients kein Systemneustart erforderlich, um die Verbindung bzw. Konfiguration mit dem DHCP-Server zu erneuern.
Stattdessen gehen die Clients automatisch in bestimmten Zeitabständen einen Neubindungszustand ein, um die Zuordnung der geleasten Adressen am DHCP-

Server zu erneuern. Dieser Vorgang wird im Hintergrund ausgeführt und ist für Sie als Anwender transparent.

Es gibt drei verschiedene Betriebsmodi eines DHCP-Servers:

- manuelle Zuordnung
  In diesem Modus werden am DHCP-Server die IP-Adressen bestimmten MAC-Adressen fest zugeordnet. Die Adressen werden der MAC-Adresse auf unbestimmte Zeit zugeteilt.
  Manuelle Zuordnungen werden vor allem dann vorgenommen, wenn der DHCP-Client unter einer festen IP-Adresse erreichbar sein soll.

- automatische Zuordnung
  Bei der automatischen Zuordnung wird am DHCP-Server ein Bereich von IP-Adressen definiert.
  Wenn die Adresse aus diesem Bereich einmal einem DHCP-Client zugeordnet wurde, dann gehört sie diesem auf unbestimmte Zeit, denn auch hier wird die zugewiesene IP-Adresse an die MAC-Adresse gebunden.

- dynamische Zuordnung
  Dieses Verfahren gleicht der automatischen Zuordnung, allerdings hat der DHCP-Server hier in seiner Konfigurationsdatei eine Angabe, wie lange eine bestimmte IP-Adresse an einen Client „vermietet“ werden darf, bevor der Client sich erneut beim Server melden und eine „Verlängerung“ beantragen muss.
  Meldet er sich nicht, wird die Adresse frei und kann an einen anderen (oder auch den gleichen) Client neu vergeben werden. Diese vom Administrator bestimmte Zeit heißt Lease-Time (zu deutsch also: „Mietzeit“).
  Manche DHCP-Server vergeben auch von der MAC-Adresse abhängige IP-Adressen, d. h. ein Client bekommt hier selbst nach längerer Netzwerkabstinenz und Ablauf der Lease-Zeit die gleiche IP-Adresse wie zuvor (es sei denn, diese ist inzwischen schon anderweitig vergeben).

Der DHCP-Client dient zur dynamischen Netzwerk-Konfiguration des Feldbuskopplers/-controllers durch Einstellung folgender Parameter:



Tabelle 85: Bedeutung der DHCP-Parameter

Parameter	Bedeutung
IP-Adresse des Clients	Netzwerk-Adresse des Feldbuskopplers/-controllers
IP-Adresse des Routers	Falls eine Kommunikation außerhalb des lokalen Netzwerkes stattfinden soll, wird die IP-Adresse des Routers (Gateway) in diesem Parameter angegeben.
Subnetmask	Die Subnetzmaske ermöglicht dem Feldbuskoppler/-controller zu unterscheiden, welche Teile der IP-Adresse das Netzwerk und welche die Netzwerkstationen bestimmen.
IP-Adressen der DNS-Server	Hier können die IP-Adressen von maximal 2 DNS-Servern angegeben werden.
Lease-Time	Hier kann die maximale Dauer definiert werden, wie lange der Feldbuskoppler/-controller die zugewiesene IP-Adresse behält. Die Höchstgrenze der Lease Time beträgt beim ETHERNET Controller 24,8 Tage. Dieses ergibt sich aus der internen Timer-Auflösung.
Renewing Time	Die Renewing Time gibt an, ab wann sich der Feldbuskoppler/-controller um die Erneuerung der Lease-Time kümmern muss.
Rebinding Time	Die Rebinding Time gibt an, nach welcher Zeit der Feldbuskoppler/-controller seine neue Adresse bekommen haben muss.

Bei Verwendung einer Lease Time, müssen die Werte für die Renewing- und Rebinding-Time auch angegeben werden. Nach Ablauf der Renewing-Time versucht der Feldbuskoppler/-controller die Lease-Time für seine IP-Adresse automatisch zu erneuern. Schlägt dieses bis zum Ablauf der Rebinding Time fehl, so versucht der Feldbuskoppler/-controller eine neue IP-Adresse zu bekommen. Die Zeit für die Renewing-Time sollte ca. die Hälfte der Lease Time betragen. Die Rebinding Time sollte ca. 7/8 der Lease Time betragen.

12.1.5.4 DNS (Domain Name Systems)

Der DNS-Client ermöglicht die Umsetzung von logischen Internet-Namen, wie z. B. www.wago.com in die entsprechende dezimale, mit Trennpunkten dargestellte IP-Adresse über einen DNS-Server. Eine umgekehrte Zuordnung ist ebenso möglich.

Die Adressen der DNS-Server werden mittels DHCP, BootP oder Web-based Management konfiguriert. Es können bis zu zwei DNS-Server angegeben werden. Die Host-Identifikation kann mit zwei Funktionen erfolgen, eine interne Host-Tabelle wird nicht unterstützt.

12.1.5.5 SNTP-Client (Simple Network Time Protocol)

Der SNTP-Client wird für die Synchronisation der Uhrzeit zwischen einem Time-Server (NTP- und SNTP-Server der Version 3 und 4) und dem im Feldbuskoppler/-controller integrierten Uhrenbaustein verwendet. Das Protokoll wird über einen UDP-Port abgearbeitet. Es wird ausschließlich die Unicast-Adressierung unterstützt.

Konfiguration des SNTP-Client

Die Konfiguration des SNTP-Client wird über das Web-based Management unter dem Link: „Clock“ vorgenommen. Folgende Parameter müssen eingestellt werden:

Tabelle 86: Bedeutung der SNTP-Parameter

Parameter	Bedeutung
Adresse des Time Servers	Die Adressvergabe kann entweder über eine IP-Adresse oder über einen Hostnamen vorgenommen werden.
Zeitzone	Für den Betrieb der ETHERNET Feldbuskoppler/-controller mit SNTP in verschiedenen Ländern muss eine Zeitzone angegeben werden. Die Einstellung der Zeitzone bezieht relativ zur GMT (Greenwich Mean Time). Es kann ein Bereich von 12 bis +12 Stunden angegeben werden.
Update Time	Die Update Time gibt das Intervall in Sekunden an, in der die Synchronisierung mit dem Time-Server erfolgen soll.
Enable Time Client	Gibt an, ob der SNTP-Client aktiviert oder deaktiviert werden soll.

12.1.5.6 FTP-Server (File Transfer Protocol)

Das File Transfer Protokoll ermöglicht es, Dateien unabhängig vom Aufbau des Betriebssystems zwischen verschiedenen Netzwerkteilnehmern auszutauschen. Bei dem ETHERNET Feldbuskoppler/-controller dient FTP dazu, die vom Anwender erstellten HTML-Seiten, das IEC-61131-Programm und den IEC-61131-Source-Code in dem (programmierbaren) Feldbuskoppler/-controller abzuspeichern und auszulesen.

Für das Dateisystem steht ein Gesamtspeicher von 1,5 MB zur Verfügung.

Das Dateisystem wird auf eine RAM-Disk abgebildet. Um die Daten der RAM-Disk permanent zu speichern, werden die Informationen zusätzlich ins Flash kopiert. Das Speichern im Flash erfolgt nach dem Schließen der Datei. Durch das Abspeichern kommt es bei Schreibzugriffen zu längeren Zugriffszeiten.

Hinweis



Zyklen für Flash auf 1 Million begrenzt!
Bis zu 1 Million Schreibzyklen sind beim Beschreiben des Flash für das Dateisystem möglich.



Die folgende Tabelle zeigt die unterstützen FTP-Kommandos für Zugriffe auf das Dateisystem:

Tabelle 87: FTP-Kommandos und deren Funktion

Kommando	Funktion
USER	Identifizierung des Anwenders
PASS	Anwenderpasswort
ACCT	Account für Zugriff auf bestimmte Dateien
REIN	Reset des Servers
QUIT	Beendet die Verbindung
PORT	Adressierung der Datenverbindung
PASV	Versetzt den Server in den Listen-Mode
TYPE	Setzt die Art der Darstellung der zu übertragenen Datei fest
STRU	Setzt die Struktur der zu übertragenen Datei fest
MODE	Setzt die Übertragungsweise der Datei fest
RETR	Datei vom Server lesen
STOR	Datei auf Server speichern
APPE	Datei auf Server speichern (Append-Modus)
ALLO	Reservierung des nötigen Speicherplatzes für die Datei
RNFR	Datei umbenennen von (mit RNTO)
RNTO	Datei umbenennen in (mit RNFR)
ABOR	laufende Funktion abbrechen
DELE	Datei löschen
CWD	Verzeichnis wechseln
LIST	Verzeichnis-Liste ausgeben
NLST	Verzeichnis-Liste ausgeben
RMD	Verzeichnis löschen
PWD	aktuellen Pfad angeben
MKD	Verzeichnis anlegen

Das TFTP (Trivial File Transfer Protocol) wird von einigen Feldbuskopplern/-controllern nicht unterstützt.

Information



Weitere Information zu den implementierten Protokollen
Die in dem Feldbuskoppler/-controller jeweils implementierten und unterstützten Protokolle sind in dem Kapitel „Technische Daten“ zu dem Feldbuskoppler/-controller aufgelistet.

12.2 SNMP (Simple Network Management Protokoll)

SNMP stellt einen Standard für das Management von Geräten in einem TCP/IP-Netzwerk dar. Es dient dem Transport von Kontrolldaten, die den Austausch von Management-Informationen, Status- und Statistikdaten zwischen einzelnen Netzwerkkomponenten und einem Management-System ermöglichen.

Eine SNMP-Management-Workstation fragt die SNMP-Agenten ab, um Informationen über die entsprechenden Geräte zu erhalten.

SNMP wird in den Versionen 1/2c und für einige Feldbuskoppler/-controller zusätzlich in der Version 3 unterstützt.

Bei SNMP in der Version 1 und 2c handelt es sich um einen Community-Nachrichtenaustausch. Dazu muss der Community-Name der Netzgemeinschaft angegeben werden.

In der Version 3 von SNMP ist der Nachrichtenaustausch an Anwender gebunden. Jedes Gerät, welches die über das WBM eingestellten Passwörter kennt, kann Werte aus dem Feldbuskoppler/-controller lesen bzw. schreiben. Bei SNMPV3 können die Nutzdaten der SNMP-Nachrichten auch verschlüsselt übertragen werden. So können die angefragten und zu schreibenden Werte nicht über ETHERNET mitgehört werden, so dass SNMPV3 häufig in sicherheitsrelevanten Netzwerken verwendet wird.

Daten eines Gerätes, auf die der SNMP-Agent zugreift oder die ein SNMP-Agent modifizieren kann, werden als SNMP-Objekt bezeichnet. Sammlungen von SNMP-Objekten sind in einer logischen Datenbank, der Management-Information-Base (MIB), enthalten, weshalb die Objekte oft auch als MIB-Objekte bezeichnet werden.

In dem Feldbuskoppler/-controller umfasst SNMP die allgemeine MIB nach RFC1213 (MIB II).

SNMP wird über den Port 161 abgearbeitet. Die Portnummer für die SNMP-Traps (Meldungen des Agenten) ist 162. Beide Ports müssen für die Nutzung von SNMP freigeschaltet sein.



12.2.1 Beschreibung der MIB II

Die Management Information Base MIB II nach RFC1213 unterteilt sich in die folgenden Gruppen:

Tabelle 88: MIB-II-Gruppen

Gruppe	Identifier
System Group	1.3.6.1.2.1.1
Interface Group	1.3.6.1.2.1.2
IP Group	1.3.6.1.2.1.4
IpRoute Table Group	1.3.6.1.2.1.4.21
ICMP Group	1.3.6.1.2.1.5
TCP Group	1.3.6.1.2.1.6
UDP Group	1.3.6.1.2.1.7
SNMP Group	1.3.6.1.2.1.11

Information



Weitere Informationen zu der MIB II
Detaillierte Informationen zu den einzelnen MIB II-Gruppen entnehmen Sie dem Kapitel „MIB-II-Gruppen“ im Anhang dieses Handbuches.

12.2.2 Beschreibung der WAGO-MIB

Die WAGO Management Information Base WAGO-MIB beschreibt WAGO-spezifische Objekte, welche über SNMP aus dem Feldbuscontroller ausgelesen werden können. Diese Objekte enthalten Informationen über den Zustand und die aktuellen Einstellungen des Gerätes.
Die WAGO-MIB liegt unter einer eigenen Enterprise-ID in der OID-Struktur von SNMP. Die Parameter der WAGO-MIB liegen unter:
iso.org.dod.internet.private.enterprise.wago (1.3.6.1.4.1.13576)

Die WAGO-MIB unterteilt sich in die folgenden Gruppen:

Tabelle 89: WAGO-MIB-Gruppen

Gruppe	Identifier
Company Group	1.3.6.1.4.1.13576.1
Product Group	1.3.6.1.4.1.13576.10
Versions Group	1.3.6.1.4.1.13576.10.1.10
Real Time Clock Group	1.3.6.1.4.1.13576.10.1.11
Ethernet Group	1.3.6.1.4.1.13576.10.1.12
Actual Error Group	1.3.6.1.4.1.13576.10.1.20
Error History Group	1.3.6.1.4.1.13576.10.1.21.1
PLC Project Group	1.3.6.1.4.1.13576.10.1.30
Http Group	1.3.6.1.4.1.13576.10.1.40.1
Ftp Group	1.3.6.1.4.1.13576.10.1.40.2
Snmp Group	1.3.6.1.4.1.13576.10.1.40.4.2
Snmp Trap String Group	1.3.6.1.4.1.13576.10.1.40.4.4
Snmp User Trap String Group	1.3.6.1.4.1.13576.10.1.40.4.5
Plc Connection Group	1.3.6.1.4.1.13576.10.1.40.5.1
Modbus Group	1.3.6.1.4.1.13576.10.1.40.6
Ethernet IP Group	1.3.6.1.4.1.13576.10.1.40.7
Process Image Group	1.3.6.1.4.1.13576.10.1.50
Plc Data Group	1.3.6.1.4.1.13576.10.1.100.1

Information



Weitere Informationen zu der WAGO-MIB
Detaillierte Informationen zu den einzelnen WAGO-MIB-Gruppen entnehmen Sie dem Kapitel „WAGO-MIB-Gruppen“ im Anhang dieses Handbuches.



12.2.3 Traps

Standard-Traps

Bei bestimmten Ereignissen sendet der SNMP-Agent selbstständig Ereignismeldungen, ohne dass diese durch den Manager angefragt werden.

Hinweis



Ereignismeldungen (Traps) im WBM freigeben!
Schalten Sie im WBM im Menü „SNMP“ unter „Trap Enable“ zunächst die Ereignismeldungen frei. Dabei können die Traps in der Version 1, 2c und 3 getrennt aktiviert werden.

Folgende Ereignismeldungen werden als Traps (SNMPv1) automatisch von dem Feldbuskoppler/-controller ausgelöst:

Tabelle 90: Standard-Traps

TrapType/TrapNummer/OID des mitgelieferten Wertes	Name	Ereignis
TrapType = 0	ColdStart	Neustart des Feldbuskopplers/-controllers
TrapType = 1	WarmStart	Reset über Service-Schalter
TrapType = 3	EthernetUp	Netzwerkverbindung gefunden
TrapType = 4	AuthenticationFailure	Unberechtigter (fehlgeschlagener) MIB-Zugriff
TrapType = 6/ ab Trap-Nummer 25 benutzerspezifisch	enterpriseSpecific	Herstellerspezifische Nachrichten und Funktionsaufruf im PFC-Programm ab Enterprise-Trap-Nummer 25

Herstellerspezifische Traps

Zusätzlich sind in dem WAGO ETHERNET TCP/IP Feldbuscontroller auch herstellerspezifische Traps definiert.

Diese werden nach der Aktivierung im WBM (bei Traps in SNMPv1 und bei Notifications in SNMPv2c/v3) bei bestimmten Ereignissen ausgelöst und automatisch versendet.

Die Traps haben alle eine eindeutige Enterprise Nummer.

Zur lesbaren Auswertung wird außerdem ein eindeutiger String angehängt. Die Strings können über SNMP verändert, bzw. in der Sprache angepasst werden. Für diese Änderungen ist in der WAGO-MIB-Datei das WAGO-spezifische Objekt „Snmp Trap String Group“ definiert (siehe hierzu das Kapitel „WAGO-MIB-Gruppen“ im Anhang).

Tabelle 91: Herstellerspezifische Traps

Enterprise-Nummer	Angehängtes MIB-Objekt	Auslösendes Ereignis
1	wioTrapKbusError	Trap, if KBus state change to error
2	wioTrapPlcStart	Trap, if PLC programm start
3	wioTrapPlcStop	Trap, if PLC programm stop
4	wioTrapPlcReset	Trap, if PLC programm reset
5	wioTrapPlcSoftwareWatchdog	Trap, if PLC software watchdog is detected
6	wioTrapPlcDivideByZero	Trap, if PLC software detect a division by zero
7	wioTrapPlcOnlineChange	Trap, if PLC programm change online
8	wioTrapPlcDownload	Trap, if PLC new programm is downloading
9	wioTrapPlcLogout	Trap, if someone logout to PLC
10	wioTrapPlcLogin	Trap, if someone login to PLC

12.2.4 Anwendungsprotokolle

Über die implementierten Anwendungsprotokolle ist mit dem Feldbuskoppler/-controller die entsprechende feldbusspezifische Kommunikation möglich. Dadurch hat der Anwender einen einfachen Zugriff von dem jeweiligen Feldbus auf den Feldbusknoten.

Die in dem Feldbuskoppler/-controller implementierten feldbusspezifischen Anwendungsprotokolle sind im Einzelnen in den nachfolgenden Kapiteln ausführlich beschrieben.



12.3 MODBUS-Funktionen

12.3.1 Allgemeines

MODBUS ist ein herstellerunabhängiger, offener Feldbusstandard für vielfältige Anwendungen in der Fertigungs- und Prozessautomation.

Das MODBUS-Protokoll ist nach dem aktuellen Internet-Draft der IETF (Internet Engineering Task Force) implementiert und erfüllt folgende Funktionen:

- Übermitteln des Prozessabbildes
- Übermitteln der Feldbusvariablen
- Übermitteln verschiedener Einstellungen und Informationen des Kopplers/Controllers über den Feldbus

Der Datentransport in der Feldebene erfolgt über TCP sowie über UDP.

Das MODBUS/TCP-Protokoll ist eine Variante des MODBUS-Protokolls, das für die Kommunikation über TCP/IP-Verbindungen optimiert wurde.

Alle Datenpakete werden über eine TCP-Verbindung mit der Portnummer 502 gesendet.

MODBUS/TCP-Datenpaket

Der allgemeine MODBUS/TCP-Header stellt sich folgendermaßen dar:

Tabelle 92: MODBUS/TCP-Header

Byte	0	1	2	3	4	5	6	7	8...n
	Kennung (wird vom Empfänger eingetragen)		Protokollkennung (immer 0 für MODBUS/TCP)		Feldlänge (Highbyte, Lowbyte)		Einheiten-kennung (Slave-Adresse)	MODBUS-Funktionscode	Daten

Information



Weitere Information
Der Telegrammaufbau ist spezifisch für die einzelnen Funktionen und deshalb detailliert in den Beschreibungen der MODBUS-Funktionscodes erläutert.

Für das MODBUS-Protokoll werden 15 Verbindungen über TCP zur Verfügung gestellt. Damit ist es möglich, von 15 Stationen zeitgleich digitale und analoge Ausgangsdaten an einem Feldbusknoten direkt auszulesen und spezielle Funktionen durch einfache MODBUS-Funktionscodes auszuführen.

Zu diesem Zweck sind eine Reihe von MODBUS-Funktionen aus der „Open MODBUS/TCP Specification“ realisiert.

Information



Weitere Information
Weiterführende Informationen zu der „Open MODBUS/TCP Specification" finden Sie im Internet unter: http://www.modbus.org

Das MODBUS-Protokoll basiert dabei im Wesentlichen auf den folgenden Grunddatentypen:

Tabelle 93: Grunddatentypen des MODBUS-Protokolls

Datentyp	Länge	Beschreibung
Discrete Inputs	1 Bit	Digitale Eingänge
Coils	1 Bit	Digitale Ausgänge
Input Register	16 Bit	Analoge Eingänge
Holding Register	16 Bit	Analoge Ausgänge

Für jeden Grunddatentyp sind ein oder mehr Funktionscodes definiert.

Mit diesen Funktionen können gewünschte binäre oder analoge Ein- und Ausgangsdaten und interne Variablen aus dem Feldbusknoten gesetzt oder direkt ausgelesen werden.



Tabelle 94: Auflistung der in dem Controller realisierten MODBUS-Funktionen

Funktionscode		Funktionsname	Zugriffsart und -beschreibung	Zugriff auf Ressourcen	
FC1	0x01	Read Coils	Lesen eines einzelnen Bit	**R:**	Prozessabbild, PFC-Variablen
FC2	0x02	Read Input Discretes	Lesen mehrerer Eingangsbits	**R:**	Prozessabbild, PFC-Variablen
FC3	0x03	Read Multiple Registers	Lesen mehrerer Eingangsregister	**R:**	Prozessabbild, PFC-Variablen, Interne Variablen, NOVRAM
FC4	0x04	Read Input Registers	Lesen mehrerer Eingangsregister	**R:**	Prozessabbild, PFC-Variablen, Interne Variablen, NOVRAM
FC5	0x05	Write Coil	Schreiben eines einzelnen Ausgangsbits	**W:**	Prozessabbild, PFC-Variablen
FC6	0x06	Write Single Register	Schreiben eines einzelnen Ausgangsregisters	**W:**	Prozessabbild, PFC-Variablen, Interne Variablen, NOVRAM
FC11	0x0B	Get Comm Event Counters	Kommunikationsereigniszähler	**R:**	Keine
FC15	0x0F	Force Multiple Coils	Schreiben mehrerer Ausgangsbits	**W:**	Prozessabbild, PFC-Variablen
FC16	0x10	Write Multiple Registers	Schreiben mehrerer Ausgangsregister	**W:**	Prozessabbild, PFC-Variablen, Interne Variablen, NOVRAM
FC22	0x16	Mask Write Register		**W:**	Prozessabbild, PFC-Variablen, NOVRAM
FC23	0x17	Read/Write Registers	Lesen und Schreiben mehrerer Ausgangsregister	**R/W:**	Prozessabbild, PFC-Variablen, NOVRAM

Um eine gewünschte Funktion auszuführen, wird der entsprechende Funktionscode und die Adresse des ausgewählten Ein- oder Ausgangskanals angegeben.

Hinweis



Bei der Adressierung auf das verwendete Zahlensystem achten!
Die aufgeführten Beispiele verwenden als Zahlenformat das Hexadezimalsystem (Bsp.: 0x000). Die Adressierung beginnt mit 0.
Je nach Software und Steuerung kann das Format und der Beginn der Adressierung variieren. Alle Adressen sind in diesem Fall dementsprechend umzurechnen.

12.3.2 Anwendung der MODBUS-Funktionen

Die grafische Übersicht zeigt anhand eines exemplarischen Feldbusknotens den Zugriff einiger MODBUS-Funktionen auf die Daten des Prozessabbildes.

DI DI AI AI DI AI DI DI AI

DO AO AO

Eingangsklemmen 750- 400 400 467 467 400 467 400 400 467

Ausgangsklemmen 750- 501 550 550

FC 3 (Read Multiple Registers)
FC 4 (Read Input Registers)

MODBUS-Adressen
0x0000 Word1
0x0001 Word2
0x0002 Word1
0x0003 Word2
0x0004 Word1
0x0005 Word2
0x0006 Word1
0x0007 Word2
0x0008 Highbyte Lowbyte

FC 1 (Read Coils)
FC 2 (Read Input Discretes)

MODBUS-Adressen
0x0000
0x0001
0x0002
0x0003
0x0004
0x0005
0x0006
0x0007
0x0008
0x0009

FC 6 (Write Single Register)
FC 16 (Write Multiple Registers)

MODBUS-Adressen
0x0000 / 0x0200 Word1
0x0001 / 0x0201 Word2
0x0002 / 0x0202 Word1
0x0003 / 0x0203 Word2
0x0004 / 0x0204 Highbyte Lowbyte

FC 3 (Read Multiple Registers)
FC 4 (Read Input Registers)

MODBUS-Adressen
0x0200 Word1
0x0201 Word2
0x0202 Word1
0x0203 Word2
0x0204 Highbyte Lowbyte

FC 5 (Write Coil)
FC 15 (Force Multiple Coils)

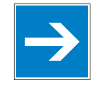
MODBUS-Adressen
0x0000 / 0x0200
0x0001 / 0x0201

FC 1 (Read Coils)
FC 2 (Read Input Discretes)

MODBUS-Adressen
0x0200
0x0201

Abbildung 83: Anwendung von MODBUS-Funktionen für einen Feldbuskoppler/-controller

Hinweis



Registerfunktionen für analoge Signale, Coil-Funktionen für binäre Signale verwenden!
Es ist sinnvoll, auf die analogen Signale mit Registerfunktionen ① und auf die binären Signale mit Coil-Funktionen ② zuzugreifen. Wird auf die binären Signale lesend oder schreibend mit Registerfunktionen ③ zugegriffen, verschieben sich die Adressen, sobald weitere analoge Busklemmen an dem Feldbuskoppler/-controller betrieben werden.



12.3.3 Beschreibung der MODBUS-Funktionen

Alle implementierten MODBUS-Funktionen werden in der folgenden Weise ausgeführt:

1. Mit der Eingabe eines Funktionscodes stellt der MODBUS/TCP-Master (z. B. ein PC) eine entsprechende Anfrage (Request) an den WAGO-Feldbusknoten.

2. Der WAGO-Feldbusknoten sendet ein Telegramm als Antwort (Response) an den Master zurück.

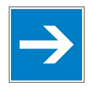
Empfängt der WAGO-Feldbusknoten eine fehlerhafte Anfrage, sendet dieser ein Fehlertelegramm (Exception) an den Master zurück. Dabei hat der im Fehlertelegramm befindliche Exception-Code die folgende Bedeutung:

Tabelle 95: Exception-Codes

Exception-Code	Bedeutung
0x01	Illegal function
0x02	Illegal data address
0x03	Illegal data value
0x04	Slave device failure
0x05	Acknowledge
0x06	Server busy
0x08	Memory parity error
0x0A	Gateway path unavailable
0x0B	Gateway target device failed to respond

In den folgenden Kapiteln wird für jeden Funktionscode der Telegrammaufbau von Request, Response und Exception mit Beispielen beschrieben.

Hinweis



Lesen und Schreiben der Ausgänge bei FC1 bis FC4 auch durch Hinzuaddieren eines Offsets möglich!
Bei den Lesefunktionen (FC1 ... FC4) können Sie zusätzlich die Ausgänge schreiben und zurücklesen, indem Sie für Adressen in dem Bereich [0_{hex} ... FF_{hex}] ein Offset von 200_{hex} (0x0200) und für Adressen in dem Bereich [6000_{hex} ... $62FC_{hex}$] ein Offset von 1000_{hex} (0x1000) zu der MODBUS-Adresse hinzu addieren.

12.3.3.1 Funktionscode FC1 (Read Coils)

Diese Funktion liest den Inhalt mehrerer Eingangs- und Ausgangsbits.

Aufbau des Request

Die Anfrage bestimmt die Startadresse und die Anzahl der zu lesenden Bits. Beispiel: Eine Anfrage, mit welcher Bit 0 bis Bit 7 gelesen werden.

Tabelle 96: Aufbau des Request für den Funktionscode FC1

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x01
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Bit count	0x0008

Aufbau der Response

Die aktuellen Werte der abgefragten Bits werden in das Datenfeld geschrieben. Eine 1 entspricht dabei dem Zustand ON und eine 0 dem Zustand OFF. Das niederwertigste Bit des ersten Datenbytes enthält das erste Bit der Anfrage. Die anderen Bits folgen aufsteigend. Falls die Anzahl der Eingänge kein Vielfaches von 8 ist, werden die verbleibenden Bits des letzten Datenbytes mit Nullen aufgefüllt.

Tabelle 97: Aufbau der Response für den Funktionscode FC1

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x01
Byte 8	Byte count	0x01
Byte 9	Bit values	0x12

Der Status der Eingänge 7 bis 0 wird als Byte-Wert 0x12 oder Binärwert 0001 0010 angezeigt. Eingang 7 ist das Bit mit dem höchsten Wert, Eingang 0 ist das Bit mit dem niedrigsten Wert dieses Bytes. Die Zuordnung erfolgt von 7 bis 0 wie folgt:

Tabelle 98: Zuordnung der Eingänge

	OFF	OFF	OFF	ON	OFF	OFF	ON	OFF
Bit	0	0	0	1	0	0	1	0
Coil	7	6	5	4	3	2	1	0



Aufbau der Exception

Tabelle 99: Aufbau der Exception für den Funktionscode FC1

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x81
Byte 8	Exception code	0x01 oder 0x02

12.3.3.2 Funktionscode FC2 (Read Input Discretes)

Diese Funktion liest den Inhalt mehrerer Eingangsbits (digitale Eingänge).

Aufbau des Request

Die Anfrage bestimmt die Startadresse und die Anzahl der zu lesenden Bits.
Beispiel: Eine Anfrage, mit welcher Bit 0 bis Bit 7 gelesen werden.

Tabelle 100: Aufbau des Request für den Funktionscode FC2

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x02
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Bit count	0x0008

Aufbau der Response

Die aktuellen Werte der abgefragten Bits werden in das Datenfeld geschrieben. Eine 1 entspricht dabei dem Zustand ON und eine 0 dem Zustand OFF. Das niederwertigste Bit des ersten Datenbytes enthält das erste Bit der Anfrage. Die anderen Bits folgen aufsteigend. Falls die Anzahl der Eingänge kein Vielfaches von 8 ist, werden die verbleibenden Bits des letzten Datenbytes mit Nullen aufgefüllt.

Tabelle 101: Aufbau der Response für den Funktionscode FC2

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x02
Byte 8	Byte count	0x01
Byte 9	Bit values	0x12

Der Status der Eingänge 7 bis 0 wird als Byte-Wert 0x12 oder Binärwert 0001 0010 angezeigt. Eingang 7 ist das Bit mit dem höchsten Wert, Eingang 0 ist das Bit mit dem niedrigsten Wert dieses Bytes. Die Zuordnung erfolgt von 7 bis 0 wie folgt:

Tabelle 102: Zuordnung der Eingänge

	OFF	OFF	OFF	ON	OFF	OFF	ON	OFF
Bit	0	0	0	1	0	0	1	0
Coil	7	6	5	4	3	2	1	0



Aufbau der Exception

Tabelle 103: Aufbau der Exception für den Funktionscode FC2

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x82
Byte 8	Exception code	0x01 oder 0x02

12.3.3.3 Funktionscode FC3 (Read Multiple Registers)

Diese Funktion dient dazu, eine Anzahl von Eingangsworten (Eingangsregister) zu lesen.

Aufbau des Request

Die Anfrage bestimmt die Adresse des Startwortes (Startregister) und die Anzahl der Register, die gelesen werden. Die Adressierung beginnt mit 0.
Beispiel: Abfrage der Register 0 und 1.

Tabelle 104: Aufbau des Request für den Funktionscode FC3

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x03
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Word count	0x0002

Aufbau der Response

Die Registerdaten der Antwort werden als 2 Bytes pro Register gepackt. Das erste Byte enthält dabei die höherwertigen Bits, das zweite Byte die niederwertigen.

Tabelle 105: Aufbau der Response für den Funktionscode FC3

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x03
Byte 8	Byte count	0x04
Byte 9, 10	Value register 0	0x1234
Byte 11, 12	Value register 1	0x2345

Aus der Antwort ergibt sich, dass Register 0 den Wert 0x1234 und Register 1 den Wert 0x2345 enthält.

Aufbau der Exception

Tabelle 106: Aufbau der Exception für den Funktionscode FC3

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x83
Byte 8	Exception code	0x01 oder 0x02



12.3.3.4 Funktionscode FC4 (Read Input Registers)

Diese Funktion dient dazu, eine Anzahl von Eingangsworten (Eingangsregister) zu lesen.

Aufbau des Request

Die Anfrage bestimmt die Adresse des Startwortes (Startregister) und die Anzahl der Register, die gelesen werden sollen. Die Adressierung beginnt mit 0.
Beispiel: Abfrage der Register 0 und 1.

Tabelle 107: Aufbau des Request für den Funktionscode FC4

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x04
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Word count	0x0002

Aufbau der Response

Die Registerdaten der Antwort werden als 2 Bytes pro Register gepackt. Das erste Byte enthält dabei die höherwertigen Bits, das zweite die niederwertigen.

Tabelle 108: Aufbau der Response für den Funktionscode FC4

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x04
Byte 8	Byte count	0x04
Byte 9, 10	Value register 0	0x1234
Byte 11, 12	Value register 1	0x2345

Aus der Antwort ergibt sich, dass Register 0 den Wert 0x1234 und Register 1 den Wert 0x2345 enthält.

Aufbau der Exception

Tabelle 109: Aufbau der Exception für den Funktionscode FC4

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x84
Byte 8	Exception code	0x01 oder 0x02

12.3.3.5 Funktionscode FC5 (Write Coil)

Diese Funktion dient dazu, ein digitales Ausgangsbit zu schreiben.

Aufbau des Request

Die Anfrage bestimmt die Adresse des Ausgangsbits. Die Adressierung beginnt mit 0.
Beispiel: Setzen des 2. Ausgangsbits (Adresse 1).

Tabelle 110: Aufbau des Request für den Funktionscode FC5

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x05
Byte 8, 9	Reference number	0x0001
Byte 10	ON/OFF	0xFF
Byte 11		0x00

Aufbau der Response

Tabelle 111: Aufbau der Response für den Funktionscode FC5

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x05
Byte 8, 9	Reference number	0x0001
Byte 10	Value	0xFF
Byte 11		0x00

Aufbau der Exception

Tabelle 112: Aufbau der Exception für den Funktionscode FC5

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01, 0x02 oder 0x03



12.3.3.6 Funktionscode FC6 (Write Single Register)

Diese Funktion schreibt einen Wert in ein einzelnes Ausgangswort (Ausgangsregister).

Aufbau des Request

Die Adressierung beginnt mit 0. Die Anfrage bestimmt die Adresse des ersten Ausgangswortes, das gesetzt werden soll. Der zu setzende Wert wird im Anfragedatenfeld bestimmt.
Beispiel: Setzen des zweiten Ausgangskanal auf den Wert 0x1234.

Tabelle 113: Aufbau des Request für den Funktionscode FC6

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x06
Byte 8, 9	Reference number	0x0001
Byte 10, 11	Register value	0x1234

Aufbau der Response

Die Antwort ist ein Echo der Anfrage.

Tabelle 114: Aufbau der Response für den Funktionscode FC6

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x06
Byte 8, 9	Reference number	0x0001
Byte 10, 11	Register value	0x1234

Aufbau der Exception

Tabelle 115: Aufbau der Exception für den Funktionscode FC6

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01 oder 0x02

12.3.3.7 Funktionscode FC11 (Get Comm Event Counter)

Diese Funktion gibt ein Statuswort und einen Ereigniszähler aus dem Kommunikationsereigniszähler des Controllers zurück. Die übergeordnete Steuerung kann mit diesem Zähler feststellen, ob der Controller die Nachrichten fehlerlos verarbeitet hat.

Nach jeder erfolgreichen Nachrichtenverarbeitung wird der Zähler hochgezählt. Fehlermeldungen oder Zählerabfragen werden nicht mitgezählt.

Aufbau des Request

Tabelle 116: Aufbau des Request für den Funktionscode FC11

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0002
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x0B

Aufbau der Response

Die Antwort enthält ein 2-Byte-Statuswort und einen 2-Byte-Ereigniszähler. Das Statuswort besteht aus Nullen.

Tabelle 117: Aufbau der Response für den Funktionscode FC11

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x0B
Byte 8, 9	Status	0x0000
Byte 10, 11	Event count	0x0003

Der Ereigniszähler zeigt, dass 3 (0x0003) Ereignisse gezählt wurden.

Aufbau der Exception

Tabelle 118: Aufbau der Exception für den Funktionscode FC11

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01 oder 0x02



12.3.3.8 Funktionscode FC15 (Force Multiple Coils)

Mit dieser Funktion wird eine Anzahl von bis zu 256 Ausgangsbits auf 1 oder 0 gesetzt.

Aufbau des Request

Das erste Bit wird mit 0 adressiert. In der Anfrage werden die Bits spezifiziert, die gesetzt werden sollen. Die geforderten 1-oder 0-Zustände werden durch die Inhalte des Anfragedatenfeldes bestimmt.

In diesem Beispiel werden 16 Bits beginnend mit Adresse 0 gesetzt. Die Anfrage enthält 2 Bytes mit dem Wert 0xA5F0 also 1010 0101 1111 0000 binär.

Das erste Byte überträgt den Wert 0xA5 an die Adresse 7 bis 0, wobei Bit 0 das niederwertigste Bit ist. Das nächste Byte überträgt den Wert 0xF0 an die Adresse 15 bis 8, wobei Bit 8 das niederwertigste Bit ist.

Tabelle 119: Aufbau des Request für den Funktionscode FC15

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0009
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x0F
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Bit count	0x0010
Byte 12	Byte count	0x02
Byte 13	Data byte1	0xA5
Byte 14	Data byte2	0xF0

Aufbau der Response

Tabelle 120: Aufbau der Response für den Funktionscode FC15

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x0F
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Bit count	0x0010

Aufbau der Exception

Tabelle 121: Aufbau der Exception für den Funktionscode FC15

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x8F
Byte 8	Exception code	0x01 oder 0x02



12.3.3.9 Funktionscode FC16 (Write Multiple Registers)

Diese Funktion schreibt Werte in eine Anzahl von Ausgangsworten (Ausgangsregister).

Aufbau des Request

Das erste Register wird mit 0 adressiert.
Die Anfragenachricht bestimmt die Register, die gesetzt werden sollen.
Pro Register werden 2 Byte an Daten gesendet.
Beispiel: Die Daten in den beiden Registern 0 und 1 werden gesetzt.

Tabelle 122: Aufbau des Request für den Funktionscode FC16

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x000B
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x10
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Word count	0x0002
Byte 12	Byte count	0x04
Byte 13, 14	Register value 1	0x1234
Byte 15, 16	Register value 2	0x2345

Aufbau der Response

Tabelle 123: Aufbau der Response für den Funktionscode FC16

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x10
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Word count	0x0002

Aufbau der Exception

Tabelle 124: Aufbau der Exception für den Funktionscode FC16

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01 oder 0x02

12.3.3.10 Funktionscode FC22 (Mask Write Register)

Diese Funktion dient dazu einzelne Bits innerhalb eines Registers zu manipulieren.

Aufbau des Request

Tabelle 125: Aufbau des Request für den Funktionscode FC22

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0002
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x16
Byte 8, 9	Reference number	0x0000
Byte 10, 11	AND mask	0x0000
Byte 12, 13	OR mask	0xAAAA

Aufbau der Response

Tabelle 126: Aufbau der Response für den Funktionscode FC22

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x10
Byte 8, 9	Reference number	0x0000
Byte 10, 11	AND mask	0x0000
Byte 12, 13	OR mask	0xAAAA

Aufbau der Exception

Tabelle 127: Aufbau der Exception für den Funktionscode FC22

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01 oder 0x02



12.3.3.11 Funktionscode FC23 (Read/Write Multiple Registers)

Diese Funktion liest Registerwerte aus und schreibt Werte in eine Anzahl von Ausgangsworten (Ausgangsregister).

Aufbau des Request

Das erste Register wird mit 0 adressiert.
Die Anfragenachricht bestimmt die Register, die gelesen und gesetzt werden sollen.
Pro Register werden 2 Byte an Daten gesendet.
Beispiel: Die Daten in dem Register 3 werden auf den Wert 0x0123 gesetzt. Aus den beiden Registern 0 und 1 werden die Werte 0x0004 und 0x5678 gelesen.

Tabelle 128: Aufbau des Request für den Funktionscode FC23

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x000F
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x17
Byte 8, 9	Reference number for read	0x0000
Byte 10, 11	Word count for read (1-125)	0x0002
Byte 12, 13	Reference number for write	0x0003
Byte 14, 15	Word count for write (1-100)	0x0001
Byte 16	Byte count (2 x word count for write)	0x02
Byte 17...(B+16)	Register values (B = Byte count)	0x0123

Aufbau der Response

Tabelle 129: Aufbau der Response für den Funktionscode FC23

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x17
Byte 8	Byte count (2 x word count for read)	0x04
Byte 9...(B+1)	Register values (B = Byte count)	0x0004 oder 0x5678

Aufbau der Exception

Tabelle 130: Aufbau der Exception für den Funktionscode FC23

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x97
Byte 8	Exception code	0x01 oder 0x02

Hinweis



Ergebnisse in überlappenden Registerbereichen sind undefiniert!
Wenn sich für das Lesen und Schreiben Registerbereiche überlappen, sind die Ergebnisse undefiniert.



12.3.4 MODBUS-Register-Mapping

In den folgenden Tabellen werden die MODBUS-Adressierung und die entsprechende IEC-61131-Adressierung für das Prozessabbild, die PFC-Variablen, die NOVRAM-Daten und die internen Variablen dargestellt.

Über die Registerdienste lassen sich die Zustände von komplexen und digitalen Busklemmen ermitteln oder verändern.

Registerzugriff Lesen (mit FC3, FC4 und FC23)

Tabelle 131: Registerzugriff Lesen (mit FC3, FC4 und FC23)

MODBUS-Adresse		IEC-61131-Adresse	Speicherbereich
[dez]	[hex]		
0...255	0x0000...0x00FF	%IW0...%IW255	Physical-Input-Area (1) First 256 Words of physical input data
256...511	0x0100...0x01FF	%QW256...%QW511	PFC-OUT-Area Flüchtige SPS-Ausgangsvariablen
512...767	0x0200...0x02FF	%QW0...%QW255	Physical-Output-Area (1) First 256 Words of physical output data
768...1023	0x0300...0x03FF	%IW256...%IW511	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
1024...4095	0x0400...0x0FFF	-	MODBUS-Exception: “Illegal data address”
4096...12287	0x1000...0x2FFF	-	Konfigurationsregister (siehe Kapitel „Konfigurationsregister“)
12288...24575	0x3000...0x5FFF	%MW0...%MW12287	NOVRAM 8 kB retain memory (max. 24 kB)
24576...25340	0x6000...0x62FC	%IW512...%IW1275	Physical-Input-Area (2) Additional 764 Words physical input data
25341...28671	0x62FD...0x6FFF	-	MODBUS-Exception: “Illegal data address”
28672...29436	0x7000...0x72FC	%QW512...%QW1275	Physical-Output-Area (2) Additional 764 Words physical output data
29437...65535	0x72FD...0xFFFF	-	MODBUS-Exception: “Illegal data address”

Registerzugriff Schreiben (mit FC6, FC16, FC22 und FC23)

Tabelle 132: Registerzugriff Schreiben (mit FC6, FC16, FC22 und FC23)

MODBUS-Adresse		IEC-61131-Adresse	Speicherbereich
[dez]	[hex]		
0...255	0x0000...0x00FF	%QW0...%QW255	Physical-Output-Area (1) First 256 Words of physical output data
256...511	0x0100...0x01FF	%IW256...%IW511	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
512...767	0x0200...0x02FF	%QW0...%QW255	Physical-Output-Area (1) First 256 Words of physical output data
768...1023	0x0300...0x03FF	%IW256...%IW511	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
1024...4095	0x0400...0x0FFF	-	MODBUS-Exception: “Illegal data address”
4096...12287	0x1000...0x2FFF	-	Konfigurationsregister (siehe Kapitel „Konfigurationsregister“)
12288...24575	0x3000...0x5FFF	%MW0...%MW12287	NOVRAM 8 kB retain memory (max. 24 kB)
24576...25340	0x6000...0x62FC	%QW512...%QW1275	Physical-Output-Area (2) Additional 764 Words physical output data
25341...28671	0x62FD...0x6FFF	-	MODBUS-Exception: “Illegal data address”
28672...29436	0x7000...0x72FC	%QW512...%QW1275	Physical-Output-Area (2) Additional 764 Words physical output data
29437...65535	0x72FD...0xFFFF	-	MODBUS-Exception: “Illegal data address”

Die digitalen MODBUS-Dienste (Coil-Dienste) sind Bitzugriffe, mit denen sich die Zustände von digitalen Busklemmen ermitteln oder verändern lassen. Komplexe Busklemmen sind mit diesen Diensten nicht erreichbar und werden ignoriert. Deshalb wird bei der Adressierung der digitalen Kanäle wieder mit 0 begonnen, so dass die MODBUS-Adresse immer identisch mit der Kanalnummer ist (der 47. digitale Eingang hat beispielsweise die MODBUS-Adresse „46“).



Bitzugriff Lesen (mit FC1 und FC2)

Tabelle 133: Bitzugriff Lesen (mit FC1 und FC2)

MODBUS-Adresse		Speicherbereich	Beschreibung
[dez]	[hex]		
0...511	0x0000...0x01FF	Physical Input Area (1)	First 512 digital inputs
512...1023	0x0200...0x03FF	Physical Output Area (1)	First 512 digital outputs
1024...4095	0x0400...0x0FFF	-	MODBUS-Exception: “Illegal data address”
4096...8191	0x1000...0x1FFF	%QX256.0...%QX511.15	PFC-OUT-Area Flüchtige SPS-Ausgangsvariablen
8192...12287	0x2000...0x2FFF	%IX256.0...%IX511.15	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
12288...32767	0x3000...0x7FFF	%MX0...%MX1279.15	NOVRAM 8 kB retain memory (max. 24 kB)
32768...34295	0x8000...0x85F7	Physical Input Area (2)	Starts with the 513th and ends with the 2039 th digital input
34296...36863	0x85F8...0x8FFF	-	MODBUS-Exception: “Illegal data address”
36864...38391	0x9000...0x95F7	Physical Output Area (2)	Starts with the 513 th and ends with the 2039 th digital output
38392...65535	0x95F8...0xFFFF	-	MODBUS-Exception: “Illegal data address”

Bitzugriff Schreiben (mit FC5 und FC15)

Tabelle 134: Bitzugriff Schreiben (mit FC5 und FC15)

MODBUS-Adresse		Speicherbereich	Beschreibung
[dez]	[hex]		
0...511	0x0000...0x01FF	Physical-Output-Area (1)	First 512 digital outputs
512...1023	0x0200...0x03FF	Physical-Output-Area (1)	First 512 digital outputs
1024...4095	0x0400...0x0FFF	-	MODBUS-Exception: “Illegal data address”
4096...8191	0x1000...0x1FFF	%IX256.0...%IX511.15	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
8192...12287	0x2000...0x2FFF	%IX256.0...%IX511.15	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
12288...32767	0x3000...0x7FFF	%MX0...%MX1279.15	NOVRAM 8 kB retain memory (max. 24 kB)
32768...34295	0x8000...0x85F7	Physical-Output-Area (2)	Starts with the 513th and ends with the 2039 th digital input
34296...36863	0x85F8...0x8FFF	-	MODBUS-Exception: “Illegal data address”
36864...38391	0x9000...0x95F7	Physical-Output-Area (2)	Starts with the 513 th and ends with the 2039 th digital output
38392...65535	0x95F8...0xFFFF	-	MODBUS-Exception: “Illegal data address”

12.3.5 MODBUS-Register

Tabelle 135: MODBUS-Register

Register-adresse	Zugriff	Länge (Wort)	Beschreibung
0x1000	R/W	1	Watchdog-Zeit lesen/schreiben
0x1001	R/W	1	Watchdog-Codiermaske 1...16
0x1002	R/W	1	Watchdog-Codiermaske 17...32
0x1003	R/W	1	Watchdog-Trigger
0x1004	R	1	Minimale Triggerzeit
0x1005	R/W	1	Watchdog stoppen (Schreibsequenz 0xAAAA, 0x5555)
0x1006	R	1	Watchdog-Status
0x1007	R/W	1	Watchdog neu starten (Schreibsequenz 0x1)
0x1008	R/W	1	Watchdog stoppen (Schreibsequenz 0x55AA oder 0xAA55)
0x1009	R/W	1	MODBUS und HTTP schließen bei Watchdog Time-out
0x100A	R/W	1	Watchdog-Konfiguration
0x100B	W	1	Watchdog-Parameter speichern
0x1020	R	1...2	LED Error-Code
0x1021	R	1	LED Error-Argument
0x1022	R	1...4	Anzahl analoger Ausgangsdaten im Prozessabbild (in Bits)
0x1023	R	1...3	Anzahl analoger Eingangsdaten im Prozessabbild (in Bits)
0x1024	R	1...2	Anzahl digitaler Ausgangsdaten im Prozessabbild (in Bits)
0x1025	R	1...4	Anzahl digitaler Eingangsdaten im Prozessabbild (in Bits)
0x1028	R/W	1	Boot-Konfiguration
0x1029	R	9	MODBUS/TCP-Statistik
0x102A	R	1	Anzahl der TCP-Verbindungen
0x1030	R/W	1	Konfiguration MODBUS/TCP-Time-out
0x1031	R	3	Lesen der MAC-ID des Kopplers/Controllers
0x1050	R	3	Diagnose angeschlossener Klemmen
0x2000	R	1	Konstante 0x0000
0x2001	R	1	Konstante 0xFFFF
0x2002	R	1	Konstante 0x1234
0x2003	R	1	Konstante 0xAAAA
0x2004	R	1	Konstante 0x5555
0x2005	R	1	Konstante 0x7FFF
0x2006	R	1	Konstante 0x8000
0x2007	R	1	Konstante 0x3FFF
0x2008	R	1	Konstante 0x4000
0x2010	R	1	Firmware-Version
0x2011	R	1	Seriencode
0x2012	R	1	Feldbuskoppler/-controller-Code
0x2013	R	1	Firmware-Versionen Major-Revision
0x2014	R	1	Firmware-Versionen Minor-Revision



Tabelle 136: MODBUS-Register (Fortsetzung)

Register-adresse	Zugriff	Länge (Wort)	Beschreibung
0x2020	R	16	Kurzbeschreibung Koppler/Controller
0x2021	R	8	Kompilierzeit der Firmware
0x2022	R	8	Kompilierdatum der Firmware
0x2023	R	32	Angabe des Firmware-Loaders
0x2030	R	65	Beschreibung der angeschlossenen Klemmen (Klemme 0...64)
0x2031	R	64	Beschreibung der angeschlossenen Klemmen (Klemme 65...129)
0x2032	R	64	Beschreibung der angeschlossenen Klemmen (Klemme 130...194)
0x2033	R	63	Beschreibung der angeschlossenen Klemmen (Klemme 195...255)
0x2040	W	1	Software-Reset (Schreibsequenz 0x55AA oder 0xAA55)
0x2041	W	1	Format Flash-Disk
0x2042	W	1	HTML-Seiten aus der Firmware extrahieren
0x2043	W	1	Werkseinstellungen

12.3.5.1 Zugriff auf Registerwerte

Um lesend oder schreibend auf Registerwerte zugreifen zu können, verwenden Sie eine beliebige MODBUS-Anwendung. Neben kommerziellen Anwendungen (beispielsweise „ModScan“) stehen Ihnen auch kostenfreie Programme zur Verfügung (siehe Internetseite http://www.modbus.org/tech.php).

Die nachfolgenden Kapitel beschreiben den Zugriff auf die Register und ihre Werte.

12.3.5.2 Watchdog-Register

Der Watchdog überwacht die Datenübertragung zwischen übergeordneter Steuerung und Feldbuskoppler/-controller. Dazu wird von der übergeordneten Steuerung eine Zeitfunktion (Time-out) in dem Controller zyklisch angestoßen.

Bei fehlerfreier Kommunikation kann diese Zeit ihren Endwert nicht erreichen, weil sie zuvor immer wieder neu gestartet wird. Läuft die Zeit jedoch ohne Unterbrechung ab, liegt ein Feldbusausfall vor.

In diesem Fall antwortet der Feldbuskoppler/-controller auf alle folgenden MODBUS-TCP/IP-Anfragen mit dem Exception-Code 0x0004 (Slave Device Failure).

Im Feldbuskoppler/-controller sind gesonderte Register für die Ansteuerung und für die Statusabfrage des Watchdogs durch die übergeordnete Steuerung vorhanden (Registeradressen 0x1000 bis 0x1008).

Nach dem Einschalten der Versorgungsspannung ist der Watchdog noch nicht aktiviert. Zunächst ist der Time-out-Wert festzulegen (Register 0x1000). Der Watchdog kann aktiviert werden, indem im Masken-Register (0x1001) ein Funktionscode geschrieben wird, der ungleich 0 ist. Eine zweite Möglichkeit zur Aktivierung besteht darin, in das Toggle-Register (0x1003) einen von 0 abweichenden Wert zu schreiben.

Durch das Lesen der minimalen Triggerzeit (Register 0x1004) wird festgestellt, ob die Watchdog-Fehlerreaktion aktiviert wurde. Falls dieser Zeitwert 0 ist, wird ein Feldbusausfall angenommen. Der Watchdog kann entsprechend der zuvor genannten beiden Möglichkeiten oder mittels Register 0x1007 neu gestartet werden.

Wenn der Watchdog einmal gestartet wurde, kann er vom Anwender aus Sicherheitsgründen lediglich über einen bestimmten Weg gestoppt werden (Register 0x1005 oder 0x1008).

Die Watchdog-Register sind analog mit den beschriebenen MODBUS-Funktionscodes (read und write) ansprechbar. Statt der Adresse eines Klemmenkanals wird dazu die jeweilige Registeradresse angegeben.

Tabelle 137: Registeradresse 0x1000

Registeradresse 0x1000 (4096$_{dez}$)	
Wert	Watchdog time, WS_TIME
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Dieses Register speichert den Wert für die Zeitüberschreitung (Time-out). Damit der Watchdog gestartet werden kann, muss der Vorgabewert auf einen Wert ungleich Null geändert werden. Die Zeit wird in Vielfachen von 100 ms gesetzt, 0x0009 bedeutet also eine Time-out-Zeit von 0.9 s. Dieser Wert kann bei laufendem Watchdog nicht geändert werden. Es gibt keinen Code, durch den der aktuelle Datenwert nochmals geschrieben werden kann, während der Watchdog aktiv ist.

Tabelle 138: Registeradresse 0x1001

Registeradresse 0x1001 (4097$_{dez}$)	
Wert	Watchdog-Funktion Codiermaske, Funktionscode 1...16, WDFCM_1_16
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Mittels dieser Maske sind die Funktionscodes einstellbar, um die Watchdog-Funktion zu triggern. Mit der „1“ kann der Funktionscode ausgewählt werden FC 1 Bit 0 FC 2 Bit 1 FC 3 Bit 0 oder 1 FC 4 Bit 2 FC 5 Bit 0 oder 2 FC 6 Bit 1 oder 2 usw. Ein Wert ungleich Null startet die Watchdog-Funktion. Wenn in die Maske ausschließlich Codes von nicht unterstützten Funktionen eingetragen werden, startet der Watchdog nicht. Ein bestehender Fehler wird zurückgesetzt und das Prozessabbild kann wieder beschrieben werden. Auch hier kann bei laufendem Watchdog keine Änderung erfolgen. Während der Watchdog aktiv ist, gibt es keinen Code, durch den der aktuelle Datenwert nochmals geschrieben werden kann.



Tabelle 139: Registeradresse 0x1002

Registeradresse 0x1002 (4098$_{dez}$)	
Wert	Watchdog-Funktion Codiermaske, Funktionscode 17...32, WD_FCM_17_32
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Gleiche Funktion wie zuvor, aber mit den Funktionscodes 17 bis 32. Diese Codes werden nicht unterstützt. Dieses Register sollte deshalb auf dem Vorgabewert belassen werden. Es gibt keinen Ausnahmecode durch den der aktuelle Datenwert nochmals geschrieben werden kann, während der Watchdog aktiv ist.

Tabelle 140: Registeradresse 0x1003

Registeradresse 0x1003 (4099$_{dez}$)	
Wert	Watchdog-Trigger, WD_TRIGGER
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Dieses Register wird für eine alternative Trigger-Methode benutzt. Durch das Schreiben unterschiedlicher Werte in dieses Register wird der Watchdog getriggert. Aufeinanderfolgende Werte müssen sich in der Größe unterscheiden. Das Schreiben eines Werts ungleich Null startet den Watchdog. Ein Watchdog-Fehler wird zurückgesetzt und das Schreiben der Prozessdaten wird wieder ermöglicht.

Tabelle 141: Registeradresse 0x1004

Registeradresse 0x1004 (4100$_{dez}$)	
Wert	Minimale aktuelle Trigger-Zeit, WD_AC_TRG_TIME
Zugang	Lesen/schreiben
Standard	0xFFFF
Beschreibung	Dieses Register speichert die aktuell kleinste Watchdog-Trigger-Zeit. Bei einem Triggern des Watchdogs, wird der gespeicherte Wert mit dem aktuellen verglichen. Ist der aktuelle Wert kleiner als der gespeicherte, wird dieser durch den aktuellen Wert ersetzt. Die Einheit ist 100 ms/Digit. Durch das Schreiben neuer Werte wird der gespeicherte Wert geändert. Dies hat keine Auswirkung auf den Watchdog. Der Wert 0x000 ist nicht erlaubt.

Tabelle 142: Registeradresse 0x1005

Registeradresse 0x1005 (4101$_{dez}$)	
Wert	Watchdog stoppen, WD_AC_STOP_MASK
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Wird der Wert 0xAAAA gefolgt von dem Wert 0x5555 in dieses Register geschrieben, stoppt der Watchdog. Die Watchdog-Fehlerreaktion wird gesperrt. Ein Watchdog-Fehler wird zurückgesetzt und das Schreiben auf die Prozessdaten wird wieder ermöglicht.

Tabelle 143: Registeradresse 0x1006

Registeradresse 0x1006 (4102$_{dez}$)	
Wert	Während Watchdog läuft, WD_RUNNING
Zugang	Lesen
Standard	0x0000
Beschreibung	Aktueller Watchdog-Status bei 0x0000: Watchdog nicht aktiv bei 0x0001: Watchdog aktiv bei 0x0002: Watchdog abgelaufen

Tabelle 144: Registeradresse 0x1007

Registeradresse 0x1007 (4103$_{dez}$)	
Wert	Watchdog neu starten, WD_RESTART
Zugang	Lesen/schreiben
Standard	0x0001
Beschreibung	Schreiben von 0x1 in das Register startet den Watchdog wieder. Wurde der Watchdog vor dem Überlauf gestoppt, wird er nicht wieder gestartet.

Tabelle 145: Registeradresse 0x1008

Registeradresse 0x1008 (4104$_{dez}$)	
Wert	Watchdog einfach anhalten, WD_AC_STOP_SIMPLE
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Durch Schreiben der Werte 0x0AA55 oder 0X55AA wird der Watchdog angehalten, falls er aktiv war. Die Watchdog-Fehlerreaktion wird vorübergehend deaktiviert. Ein anstehender Watchdog-Fehler wird zurückgesetzt und ein Schreiben ins Watchdog-Register ist wieder möglich.

Tabelle 146: Registeradresse 0x1009

Registeradresse 0x1009 (4105$_{dez}$)	
Wert	MODBUS-Socket nach Watchdog-Time-out schließen
Zugang	Lesen/schreiben
Beschreibung	0: MODBUS-Socket wird nicht geschlossen 1: MODBUS-Socket wird geschlossen

Tabelle 147: Registeradresse 0x100A

Registeradresse 0x100A (4106$_{dez}$)	
Wert	Alternativer Watchdog
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Schreiben eines Zeitwertes in Register 0x1000 Register 0x100A = 0x0001: Watchdog wird aktiv geschaltet Mit dem ersten MODBUS-Telegramm wird der Watchdog gestartet. Der Watchdog wird mit jedem MODBUS/TCP-Befehl getriggert. Nach Ablauf der Watchdog-Zeit werden alle Ausgänge auf Null gesetzt. Die Ausgänge können durch erneutes Schreiben wieder gesetzt werden. Das Register 0x00A ist remanent und damit auch das Register 0x1000. Bei eingeschaltetem Watchdog lässt sich der Zeitwert in Register 0x1000 nicht mehr ändern.



Die Länge jedes Registers beträgt 1 Wort, d. h. bei jedem Zugriff kann lediglich ein Wort geschrieben oder gelesen werden. Im Folgenden werden zwei Beispiele zum Setzen des Wertes für die Zeitüberschreitung aufgeführt:

Watchdog für eine Zeitüberschreitung von 1 Sekunde oder mehr setzen:

1. Schreiben Sie 0x000A in das Register für Zeitüberschreitung (0x1000). (Register 0x1000 arbeitet mit Vielfachen von 100 ms; 1 s = 1000 ms; 1000 ms / 100 ms = 10_{dez} = A_{hex})

2. Starten Sie mittels des Funktionscodes 5 den Watchdog, indem Sie 0x0010 ($=2^{(5-1)}$) in die Codiermaske (Register 0x1001) schreiben.

Tabelle 148: Watchdog starten

FC	FC16	FC15	FC14	FC13	FC12	FC11	FC10	FC9	FC8	FC7	FC6	**FC5**	FC4	FC3	FC2	FC1
Bit	15	14	13	12	11	10	9	8	7	6	5	**4**	3	2	1	0
bin	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0
hex	0				0				**1**				0			

Der Funktionscode 5 (Schreiben eines digitalen Ausgangsbits) triggert den Watchdog kontinuierlich, um den Watchdog-Timer innerhalb der angegebenen Zeit immer wieder neu zu starten. Wird zwischen den Anfragen mehr als 1 Sekunde erreicht, ist ein Watchdog-Time-out-Fehler aufgetreten.

3. Um den Watchdog zu stoppen, schreiben Sie den Wert 0x0AA55 oder 0X55AA in das Register 0x1008 (Watchdog einfach anhalten, WD_AC_STOP_SIMPLE).

Watchdog für eine Zeitüberschreitung von 10 Minuten oder mehr setzen

1. Schreiben Sie 0x1770 (= 10*60*1000 ms / 100 ms) in das Register für Zeitüberschreitung (0x1000). (Register 0x1000 arbeitet mit Vielfachen von 100 ms; 10 min = 600.000 ms; 600.000 ms / 100 ms = 6000_{dez} = 1770_{hex})

2. Starten Sie den Watchdog, indem Sie 0x0001 in den Watchdog-Trigger-Register (0x1003) schreiben.

3. Um den Watchdog zu triggern, schreiben Sie unterschiedliche Werte, z. B. Zählwerte 0x0000, 0x0001 etc. in das Watchdog-Trigger-Register (0x1003).

Die nacheinander geschriebenen Werte müssen sich in der Größe unterscheiden. Das Schreiben eines Wertes ungleich Null startet den Watchdog. Watchdog-Fehler werden zurückgesetzt und das Schreiben der Prozessdaten wird wieder ermöglicht.

4. Um den Watchdog zu stoppen, schreiben Sie den Wert 0x0AA55 oder 0X55AA in das Register 0x1008 (Watchdog einfach anhalten, WD_AC_STOP_SIMPLE).



Tabelle 149: Registeradresse 0x100B

Registeradresse 0x100B (4107$_{dez}$)	
Wert	Save-Watchdog-Parameter
Zugang	Schreiben
Standard	0x0000
Beschreibung	Mit Schreiben von „1“ in Register 0x100B werden die Register 0x1000, 0x1001, 0x1002 auf „remanent“ gesetzt.



12.3.5.3 Diagnoseregister

Folgende Register können gelesen werden, um einen Fehler des Feldbusknotens zu bestimmen:

Tabelle 150: Registeradresse 0x1020

Registeradresse 0x1020 (4128$_{dez}$)	
Wert	LedErrCode
Zugang	Lesen
Beschreibung	Angabe des Fehlercodes

Tabelle 151: Registeradresse 0x1021

Registeradresse 0x1021 (4129$_{dez}$)	
Wert	LedErrArg
Zugang	Lesen
Beschreibung	Angabe des Fehlerargumentes

12.3.5.4 Konfigurationsregister

Folgende Register können gelesen werden, um die Konfiguration der angeschlossenen Klemmen zu bestimmen:

Tabelle 152: Registeradresse 0x1022

Registeradresse 0x1022 (4130$_{dez}$)	
Wert	CnfLen.AnalogOut
Zugang	Lesen
Beschreibung	Anzahl E/A-Bits bei den Prozessdatenworten der Ausgänge

Tabelle 153: Registeradresse 0x1023

Registeradresse 0x1023 (4131$_{dez}$)	
Wert	CnfLen.AnalogInp
Zugang	Lesen
Beschreibung	Anzahl E/A-Bits bei den Prozessdatenworten der Eingänge

Tabelle 154: Registeradresse 0x1024

Registeradresse 0x1024 (4132$_{dez}$)	
Wert	CnfLen.DigitalOut
Zugang	Lesen
Beschreibung	Anzahl E/A-Bits bei den Prozessdatenbits der Ausgänge

Tabelle 155: Registeradresse 0x1025

Registeradresse 0x1025 (4133$_{dez}$)	
Wert	CnfLen.DigitalInp
Zugang	Lesen
Beschreibung	Anzahl E/A-Bits bei den Prozessdatenbits der Eingänge

Tabelle 156: Registeradresse 0x1028

Registeradresse 0x1028 (4136$_{dez}$)	
Wert	Bootoptions
Zugang	Lesen/schreiben
Beschreibung	Bootkonfiguration: 1: BootP 2: DHCP 4: EEPROM



Tabelle 157: Registeradresse 0x1029

Registeradresse 0x1029 (4137$_{dez}$) mit bis zu 9 Worten		
Wert	MODBUS/TCP-Statistik	
Zugang	Lesen/schreiben	
Beschreibung	1 Wort SlaveDeviceFailure	→ Klemmenbusfehler, Feldbusfehler bei eingeschaltetem Watchdog
	1 Wort BadProtocol	→ Fehler im MODBUS/TCP-Header
	1 Wort BadLength	→ Falsche Telegrammlänge
	1 Wort BadFunction	→ Ungültiger Funktionscode
	1 Wort Bad Address	→ Ungültige Registeradresse
	1 Wort BadData	→ Ungültiger Wert
	1 Wort TooManyRegisters	→ Anzahl der zu bearbeitenden Register zu groß, Lesen/Schreiben 125/100
	1 Wort TooManyBits	→ Anzahl der zu bearbeitenden Coils zu groß, Lesen/Schreiben 2000/800
	1 Wort ModTcpMessageCounter	→ Anzahl der empfangenen MODBUS/TCP-Telegramme
	Durch Schreiben von 0xAA55 oder 0x55AA wird das Register zurückgesetzt.	

Tabelle 158: Registeradresse 0x102A

Registeradresse 0x102A (4138$_{dez}$) mit bis zu 1 Wort	
Wert	MODBUS/TCP-Connections
Zugang	Lesen
Beschreibung	Anzahl der TCP-Verbindungen

Tabelle 159: Registeradresse 0x1030

Registeradresse 0x1030 (4144$_{dez}$) mit bis zu 1 Wort	
Wert	Konfiguration MODBUS/TCP-Time-out
Zugang	Lesen/schreiben
Standard	0x0258 (600 dezimal)
Beschreibung	Dieses Register speichert den Wert für eine TCP-Verbindungsüberwachung. Der Standardwert ist 600 ms (60 Sekunden), die Zeitbasis ist 100 ms, der Minimalwert ist 100 ms. Geöffnete TCP-Verbindungen werden automatisch geschlossen, wenn die eingetragene Zeit je Verbindung überschritten wurde. Wird der Wert auf ‚0' gesetzt, ist der Watchdog nicht aktiv. Der Watchdog wird mit einer Anfrage auf der Verbindung getriggert.

Tabelle 160: Registeradresse 0x1031

Registeradresse 0x1031 (4145$_{dez}$) mit bis zu 3 Worten	
Wert	Lesen der MAC-ID des Kopplers/Controllers
Zugang	Lesen
Beschreibung	Ausgabe der MAC-ID, Länge 3 Worte

Tabelle 161: Registeradresse 0x1050

Registeradresse 0x1050 (4176$_{dez}$)	**ab Firmwarestand 9**
Wert	Diagnose angeschlossener Klemmen
Zugang	Lesen
Beschreibung	Diagnose angeschlossener Klemmen, Länge 3 Worte Wort 1: Klemmennummer Wort 2: Kanalnummer Wort 3: Diagnose

Tabelle 162: Registeradresse 0x2030

Registeradresse 0x2030 (8240$_{dez}$) mit bis zu 65 Worten	
Wert	Beschreibung der angeschlossenen Klemmen
Zugang	Lesen der Klemmen 0...64
Beschreibung	Länge 1...65 Worte Über Register 0x2030 kann die Konfiguration des Knotens ermittelt werden. Dabei wird die Artikelnummer der Klemmen bzw. des Kopplers/Controllers (ohne führende 750) der Reihe nach aufgelistet. Jede Bezeichnung wird in einem Wort dargestellt. Da Artikelnummern von digitalen Klemmen nicht ausgelesen werden können, wird eine digitale Klemme codiert dargestellt. Die einzelnen Bits haben dann die folgende Bedeutung: Bitposition 0 → Eingangsklemme Bitposition 1 → Ausgangsklemme Bitposition 2...7 → nicht benutzt Bitposition 8...14 → Klemmengröße in Bit Bitposition 15 → Kennung digitale Klemme

Beispiele:																
4-Kanal-Digitaleingangsklemme = 0x8401																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Code	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1
Hex	8				4				0				1			
2-Kanal-Digitalausgangsklemme = 0x8202																
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Code	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
Hex	8				2				0				2			

Tabelle 163: Registeradresse 0x2031

Registeradresse 0x2031 (8241$_{dez}$) mit bis zu 65 Worten	
Wert	Beschreibung der angeschlossenen Klemmen
Zugang	Lesen der Klemmen 65...128
Beschreibung	Länge 1...64 Worte Über Register 0x2031 kann die Konfiguration des Knotens ermittelt werden. Dabei wird die Artikelnummer der Klemmen bzw. des Kopplers/Controllers (ohne führende 750) der Reihe nach aufgelistet. Jede Bezeichnung wird in einem Wort dargestellt. Da Artikelnummern von digitalen Klemmen nicht ausgelesen werden können, wird eine digitale Klemme codiert dargestellt. Die einzelnen Bits haben dann die folgende Bedeutung : Bitposition 0 → Eingangsklemme Bitposition 1 → Ausgangsklemme Bitposition 2...7 → nicht benutzt Bitposition 8...14 → Klemmengröße in Bit Bitposition 15 → Kennung digitale Klemme



Tabelle 164: Registeradresse 0x2032

Registeradresse 0x2032 (8242$_{dez}$) mit bis zu 65 Worten	
Wert	Beschreibung der angeschlossenen Klemmen
Zugang	Lesen der Klemmen 129...192
Beschreibung	Länge 1...64 Worte Über Register 0x2032 kann die Konfiguration des Knotens ermittelt werden. Dabei wird die Artikelnummer der Klemmen bzw. des Kopplers/Controllers (ohne führende 750) der Reihe nach aufgelistet. Jede Bezeichnung wird in einem Wort dargestellt. Da Artikelnummern von digitalen Klemmen nicht ausgelesen werden können, wird eine digitale Klemme codiert dargestellt. Die einzelnen Bits haben dann die folgende Bedeutung: Bitposition 0 → Eingangsklemme Bitposition 1 → Ausgangsklemme Bitposition 2...7 → nicht benutzt Bitposition 8...14 → Klemmengröße in Bit Bitposition 15 → Kennung digitale Klemme

Tabelle 165: Registeradresse 0x2033

Registeradresse 0x2033 (8243$_{dez}$) mit bis zu 65 Worten	
Wert	Beschreibung der angeschlossenen Klemmen
Zugang	Lesen der Klemmen 193...255
Beschreibung	Länge 1...63 Worte Über Register 0x2033 kann die Konfiguration des Knotens ermittelt werden. Dabei wird die Artikelnummer der Klemmen bzw. des Kopplers/Controllers (ohne führende 750) der Reihe nach aufgelistet. Jede Bezeichnung wird in einem Wort dargestellt. Da Artikelnummern von digitalen Klemmen nicht ausgelesen werden können, wird eine digitale Klemme codiert dargestellt. Die einzelnen Bits haben dann die folgende Bedeutung: Bitposition 0 → Eingangsklemme Bitposition 1 → Ausgangsklemme Bitposition 2...7 → nicht benutzt Bitposition 8...14 → Klemmengröße in Bit Bitposition 15 → Kennung digitale Klemme

Tabelle 166: Registeradresse 0x2040

Registeradresse 0x2040 (8256$_{dez}$)	
Wert	Ausführen eines Software-Resets
Zugang	Schreiben (Schreibsequenz 0xAA55 oder 0x55AA)
Beschreibung	Durch Schreiben der Werte 0xAA55 oder 0x55AA führt der Feldbuskoppler/-controller einen Neustart durch.

Tabelle 167: Registeradresse 0x2041

Registeradresse 0x2041 (8257$_{dez}$)	**ab Firmwarestand 3**
Wert	Flash-Format
Zugang	Schreiben (Schreibsequenz 0xAA55 oder 0x55AA)
Beschreibung	Das Flash-Dateisystem wird neu formatiert

Tabelle 168: Registeradresse 0x2042

Registeradresse 0x2042 (8258$_{dez}$)	**ab Firmwarestand 3**
Wert	Dateien extrahieren
Zugang	Schreiben (Schreibsequenz 0xAA55 oder 0x55AA)
Beschreibung	Die Standarddateien (HTML-Seiten) des Kopplers/Controllers werden extrahiert und in das Flash geschrieben.

Tabelle 169: Registeradresse 0x2043

Registeradresse 0x2043 (8259$_{dez}$)	**ab Firmwarestand 9**
Wert	0x55AA
Zugang	Schreiben
Beschreibung	Werkseinstellungen



12.3.5.5 Firmware-Informationsregister

Folgende Register werden genutzt, um Informationen zur Firmware des Kopplers/Controllers auszulesen:

Tabelle 170: Registeradresse 0x2010

Registeradresse 0x2010 (8208$_{dez}$) mit bis zu 1 Wort	
Wert	Revision, INFO_REVISION
Zugang	Lesen
Beschreibung	Firmware-Index, z. B. 0005 für Version 5

Tabelle 171: Registeradresse 0x2011

Registeradresse 0x2011 (8209$_{dez}$) mit bis zu 1 Wort	
Wert	Series code, INFO_SERIES
Zugang	Lesen
Beschreibung	WAGO-Baureihennummer, z. B. 0750 für WAGO-I/O-SYSTEM 750

Tabelle 172: Registeradresse 0x2012

Registeradresse 0x2012 (8210$_{dez}$) mit bis zu 1 Wort	
Wert	Item number, INFO_ITEM
Zugang	Lesen
Beschreibung	WAGO-Bestellnummer, z. B. 841 für den Controller 750-841, 341 für den Koppler 750-341 etc.

Tabelle 173: Registeradresse 0x2013

Registeradresse 0x2013 (8211$_{dez}$) mit bis zu 1 Wort	
Wert	Major sub item code, INFO_MAJOR
Zugang	Lesen
Beschreibung	Firmware-Version Major-Revision

Tabelle 174: Registeradresse 0x2014

Registeradresse 0x2014 (8212$_{dez}$) mit bis zu 1 Wort	
Wert	Minor sub item code, INFO_MINOR
Zugang	Lesen
Beschreibung	Firmware-Version Minor-Revision

Tabelle 175: Registeradresse 0x2020

Registeradresse 0x2020 (8224$_{dez}$) mit bis zu 16 Worten	
Wert	Description, INFO_DESCRIPTION
Zugang	Lesen
Beschreibung	Informationen zum Feldbuskoppler/-controller, 16 Worte

Tabelle 176: Registeradresse 0x2021

Registeradresse 0x2021 (8225$_{dez}$) mit bis zu 8 Worten	
Wert	Description, INFO_DESCRIPTION
Zugang	Lesen
Beschreibung	Zeit des Firmwarestandes, 8 Worte

Tabelle 177: Registeradresse 0x2022

Registeradresse 0x2022 (8226$_{dez}$) mit bis zu 8 Worten	
Wert	Description, INFO_DATE
Zugang	Lesen
Beschreibung	Datum des Firmwarestandes, 8 Worte

Tabelle 178: Registeradresse 0x2023

Registeradresse 0x2023 (8227$_{dez}$) mit bis zu 32 Worten	
Wert	Description, INFO_LOADER_INFO
Zugang	Lesen
Beschreibung	Info über Programmierung der Firmware, 32 Worte



12.3.5.6 Konstantenregister

Folgende Register enthalten Konstanten, die genutzt werden können, um die Kommunikation mit dem Master zu testen:

Tabelle 179: Registeradresse 0x2000

Registeradresse 0x2000 (8192$_{dez}$)	
Wert	Null, GP_ZERO
Zugang	Lesen
Beschreibung	Konstante mit Null

Tabelle 180: Registeradresse 0x2001

Registeradresse 0x2001 (8193$_{dez}$)	
Wert	Einsen, GP_ONES
Zugang	Lesen
Beschreibung	Konstante mit Einsen. • „-1“, wenn Konstante als „signed int“ deklariert ist • “MAXVALUE“, wenn Konstante als „unsigned int“ deklariert ist

Tabelle 181: Registeradresse 0x2002

Registeradresse 0x2002 (8194$_{dez}$)	
Wert	1,2,3,4, GP_1234
Zugang	Lesen
Beschreibung	Konstanter Wert, zum Testen, ob High- und Low-Byte getauscht sind (Intel/Motorola Format). Sollte im Master als 0x1234 erscheinen. Erscheint 0x3412, müssen High- und Low-Byte getauscht werden.

Tabelle 182: Registeradresse 0x2003

Registeradresse 0x2003 (8195$_{dez}$)	
Wert	Maske 1, GP_AAAA
Zugang	Lesen
Beschreibung	Konstante, die anzeigt, ob alle Bits vorhanden sind. Wird zusammen mit Register 0x2004 genutzt.

Tabelle 183: Registeradresse 0x2004

Registeradresse 0x2004 (8196$_{dez}$)	
Wert	Maske 1, GP_5555
Zugang	Lesen
Beschreibung	Konstante, die anzeigt, ob alle Bits vorhanden sind. Wird zusammen mit Register 0x2003 genutzt.

Tabelle 184: Registeradresse 0x2005

Registeradresse 0x2005 (8197$_{dez}$)	
Wert	Größte positive Zahl, GP_MAX_POS
Zugang	Lesen
Beschreibung	Konstante, um die Arithmetik zu kontrollieren

Tabelle 185: Registeradresse 0x2006

Registeradresse 0x2006 (8198$_{dez}$)	
Wert	Größte negative Zahl, GP_MAX_NEG
Zugang	Lesen
Beschreibung	Konstante, um die Arithmetik zu kontrollieren

Tabelle 186: Registeradresse 0x2007

Registeradresse 0x2007 (8199$_{dez}$)	
Wert	Größte halbe positive Zahl, GP_HALF_POS
Zugang	Lesen
Beschreibung	Konstante, um die Arithmetik zu kontrollieren

Tabelle 187: Registeradresse 0x2008

Registeradresse 0x2008 (8200$_{dez}$)	
Wert	Größte halbe negative Zahl, GP_HALF_NEG
Zugang	Lesen
Beschreibung	Konstante, um die Arithmetik zu kontrollieren

Tabelle 188: Registeradresse 0x3000 bis 0x5FFF

Registeradresse 0x3000 bis 0x5FFF (12288$_{dez}$ bis 24575$_{dez}$)	
Wert	Retain-Bereich
Zugang	Lesen/schreiben
Beschreibung	In diesen Registern kann auf den Merker/Retain-Bereich zugegriffen werden.

12.4 EtherNet/IP (Ethernet/Industrial Protocol)

12.4.1 Allgemeines

EtherNet/IP steht für "Ethernet Industrial Protocol" und definiert einen offenen Industrie-Standard, der das klassische Ethernet mit einem Industrie-Protokoll erweitert. Dieser Standard wurde gemeinsam von der „ControlNet International“ (CI) und der „Open DeviceNet Vendor Association“ (ODVA) entwickelt mit Unterstützung der „Industrial Ethernet Association“ (IEA).

Durch dieses Kommunikationssystem wird es Geräten ermöglicht, zeitkritische Applikationsdaten in einer industriellen Umgebung auszutauschen. Das Gerätespektrum reicht von einfachen I/O-Geräten (z. B. Sensoren) bis zu komplexen Steuerungen (z. B. Roboter).

EtherNet/IP basiert auf der TCP/IP-Protokoll-Familie und übernimmt somit die unteren 4 Schichten des OSI-7-Schichten-Modells in unveränderter Form, so dass alle Standard-Ethernet-Kommunikationsmodule, wie z. B. Interface-Karten für PC, Kabel, Konnektoren, Hubs und Switches mit EtherNet/IP gleichfalls verwendet werden können.
Oberhalb der Transport-Schicht befindet sich das „Encapsulation Protocol“, mit dem das „Common Industrial Protocol“ (CIP) auf TCP/IP und UDP/IP aufgesetzt ist.



CIP, als ein großer netzwerkunabhängiger Standard, wird bereits bei ControlNet und DeviceNet benutzt. Die Überführung einer Applikation auf eines dieser Systeme ist somit sehr einfach realisierbar. Der Datenaustausch basiert auf einem Objektmodell.

ControlNet, DeviceNet und EtherNet/IP haben auf diese Weise dasselbe Applikationsprotokoll und können deshalb gemeinsam Geräteprofile und Objekt-Libraries nutzen. Diese Objekte machen eine Plug-and-play-Interoperabilität zwischen komplexen Geräten verschiedener Hersteller möglich.

12.4.2 Protokollübersicht im OSI-Modell

Zur Verdeutlichung der Gemeinsamkeiten zwischen DeviceNet, ControlNet und EtherNet/IP zeigt die folgende Darstellung die Einordnung der Protokolle in das 7-schichtige OSI-Referenzmodell (Open Systems Interconnection Reference Model).

Tabelle 189: OSI-Referenzmodell

<table>
<tr><td>7
Application Layer</td><td colspan="3">Object Libary
(Communications, Applications, Time Synchronization)</td><td>Safety Object Libary</td><td rowspan="3">Comon Industrial Protocol (CIP)</td></tr>
<tr><td>6
Presentation Layer</td><td colspan="3">Data Management Services
Explicit and I/O Messages</td><td>Safety Services and Messages</td></tr>
<tr><td>5
Session Layer</td><td colspan="4">Connection Management, Routing</td></tr>
<tr><td>4
Transport Layer</td><td>TCP/UDP</td><td rowspan="2">CompoNet
Network and Transport</td><td rowspan="2">ControlNet
Network
and
Transport</td><td rowspan="2">DeviceNet
Network and
Transport</td><td rowspan="4">Network Adaptations of CIP</td></tr>
<tr><td>3
Network Layer</td><td>Internet Protocol</td></tr>
<tr><td>2
Data Link Layer</td><td>Ethernet
CSMA/CD</td><td>CompoNet
Time Slot</td><td>ControlNet
CTDMA</td><td>CAN
CSMA/NBA</td></tr>
<tr><td>1
Physical Layer</td><td>Ethernet</td><td>CompoNet</td><td>ControlNet</td><td>DeviceNet</td></tr>
</table>



12.4.3 Eigenschaften der EtherNet/IP-Protokollsoftware

Die Ethernet/IP-Produktklassen sind in insgesamt 4 Level aufgeteilt, wobei jeder eine gewisse Funktionalität beinhaltet. Jeder höhere Level wiederum beinhaltet mindestens eine Funktionalität eines niedrigeren Levels. Der Feldbuskoppler unterstützt die Level 1 und 2 der Ethernet/IP-Produktklassen, die unmittelbar aufeinander aufbauen.

Level 2
Level 1

Level 2: Level 1 + I/O Messages – Server

Level 1: Explicit Messages – Server

- UCMM fähig (verbindungslos, Client und Server)
- 128 „Encapsulation Protocol Sessions“
- 128 Klasse 3-Verbindungen oder Klasse 1 (kombiniert)

Klasse 3-Verbindung – explizite Nachrichten
(verbindungsorientiert, Client und Server)

Klasse 1-Verbindung – I/O Nachrichten
(verbindungsorientiert, Client und Server)

12.4.4 EDS-Datei

Die „Electronic Data Sheets“-Datei, kurz: EDS-Datei, enthält die Kenndaten des Feldbuskopplers/-controllers und Angaben zu seinen Kommunikationsfähigkeiten. Die für den EtherNet/IP-Betrieb benötigte EDS-Datei wird von der jeweiligen Projektierungssoftware eingelesen bzw. installiert.

Hinweis



Download der EDS-Datei!
Sie erhalten die EDS-Datei im Download-Bereich der WAGO-Internetseiten unter:
www.wago.com → Service → Downloads → AUTOMATION

Information



Information zur Installation der EDS-Datei
Entnehmen Sie bitte Hinweise zur Installation der EDS-Datei der Dokumentation zu der von Ihnen genutzten Projektierungssoftware.

12.4.5 Objektmodell

12.4.5.1 Allgemeines

Für die Netzwerkkommunikation verwendet EtherNet/IP ein Objektmodell, in dem alle Funktionen und Daten eines Gerätes beschrieben sind.
Jeder Knoten im Netz wird als Sammlung von Objekten dargestellt.
Das Objektmodell enthält Begriffe, die folgendermaßen definiert sind:

Objekt (object):
Ein Objekt ist eine abstrakte Darstellung von einzelnen, zusammengehörigen Bestandteilen innerhalb eines Gerätes. Es ist bestimmt durch seine Daten oder Eigenschaften (Attributes), seine nach außen bereitgestellten Funktionen oder Dienste (Services) und durch sein definiertes Verhalten (Behaviour).

Klasse (class):
Eine Klasse beschreibt eine Reihe von Objekten, die alle die gleiche Art von Systemkomponenten darstellen. Eine Klasse ist die Verallgemeinerung eines Objektes. Alle Objekte in einer Klasse sind in Bezug auf ihre Form und ihr Verhalten identisch, wobei sie jedoch unterschiedliche Attributwerte umfassen können.

Instanz (instance):
Eine Instanz beschreibt eine spezifische und tatsächliche (physikalische) Ausprägung eines Objektes. Die Benennungen „Objekt", „Instanz" und „Objektinstanz" beziehen sich alle auf eine spezifische Instanz.
Unterschiedliche Instanzen einer Klasse haben die gleichen Dienste (services), das gleiche Verhalten (behaviour) und die gleichen Variablen (attributes). Sie können jedoch unterschiedliche Variablenwerte haben.
Beispiel: Eine Instanz der Objektklasse „Land" ist beispielsweise Finnland.

Variable (attribute):
Die Variablen (attributes) beschreiben ein externes sichtbares Merkmal oder die Funktion eines Objektes. Typische Attribute sind beispielsweise Konfigurations- oder Statusinformationen.
Beispiel: Es wird der ASCII-Name eines Objektes oder die Wiederholungsfrequenz eines periodischen Objektes ausgegeben.

Dienst (service):
Ein Dienst ist eine Funktion, die von einem Objekt und/oder einer Objekt-Klasse unterstützt wird. CIP definiert eine Gruppe gemeinsamer Dienste, die auf die Variablen (Attribute) angewendet werden. Diese Dienste führen festgelegte Aktionen durch.
Beispiel: Das Lesen von Variablen.

Verhalten (behaviour):
Das Verhalten legt fest, wie ein Objekt funktioniert. Die Funktionen resultieren aus unterschiedlichen Ereignissen, die das Objekt ermittelt, wie zum Beispiel der Empfang von Serviceanforderungen, die Erfassung interner Störungen oder der Ablauf von Zeitnehmern.



12.4.5.2 Klassen-Übersicht

Die CIP-Klassen sind in der CIP-Spezifikation der ODVA enthalten. Sie beschreiben, unabhängig von der physikalischen Schnittstelle, z. B. Ethernet, CAN, deren Eigenschaften (Band 1 „Common Industrial Protocol“). Die physikalische Schnittstelle wird in einer weiteren Spezifikation beschrieben. Für EtherNet/IP ist das der Band 2 („EtherNet/IP Adaption of CIP“), welcher die Anpassung des EtherNet/IP an CIP beschreibt.

WAGO nutzt hierbei die Klassen 01_{hex}, 02_{hex}, 04_{hex}, 05_{hex}, 06_{hex} und $F4_{hex}$, welche in Band 1 („Common Industrial Protocol“) beschrieben sind.
Aus dem Band 2 („EtherNet/IP Adaption of CIP“) werden die Klassen $F5_{hex}$ und $F6_{hex}$ unterstützt.

Darüber hinaus stehen WAGO-spezifische Klassen zur Verfügung, die in der unten stehenden Übersichtstabelle aufgeführt sind.

Alle gelisteten CIP-Common-Klassen und im Anschluss daran die WAGO-spezifischen Klassen werden, nach einer kurzen Erläuterung der Tabellenköpfe in den Objektbeschreibungen, in den folgenden einzelnen Kapiteln näher beschrieben.

Tabelle 190: Übersicht CIP-Common-Klassen

Klasse	Name
01_{hex}	Identity
02_{hex}	Message Router
04_{hex}	Assembly
05_{hex}	Connection
06_{hex}	Connection Manager
$F4_{hex}$	Port Class Object
$F5_{hex}$	TCP/IP Interface Object
$F6_{hex}$	Ethernet Link Object

Tabelle 191: Übersicht WAGO-spezifische Klassen

Klasse	Name
64 hex	Coupler/Controller Configuration Object
65 hex	Discrete Input Point
66 hex	Discrete Output Point
67 hex	Analog Input Point
68 hex	Analog Output Point
69 hex	Discrete Input Point Extended 1
6A hex	Discrete Output Point Extended 1
6B hex	Analog Input Point Extended 1
6C hex	Analog Output Point Extended 1
6D hex	Discrete Input Point Extended 2
6E hex	Discrete Output Point Extended 2
6F hex	Analog Input Point Extended 2
70 hex	Analog Output Point Extended 2
71 hex	Discrete Input Point Extended 3
72 hex	Discrete Output Point Extended 3
73 hex	Analog Input Point Extended 3
74 hex	Analog Output Point Extended 3
80 hex	Module Configuration
81 hex	Module Configuration Extended 1
A0 hex	Input fieldbus variable USINT
A1 hex	Input fieldbus variable USINT Extended 1
A2 hex	Input fieldbus variable USINT Extended 2
A3 hex	Output fieldbus variable USINT
A4 hex	Output fieldbus variable USINT Extended 1
A5 hex	Output fieldbus variable USINT Extended 2
A6 hex	Input fieldbus variable UINT
A7 hex	Input fieldbus variable UINT Extended 1
A8 hex	Output fieldbus variable UINT
A9 hex	Output fieldbus variable UINT Extended 1
AA hex	Input fieldbus variable UDINT
AB hex	Input fieldbus variable UDINT Offset UINT
AC hex	Output fieldbus variable UDINT
AD hex	Output fieldbus variable UDINT Offset UINT



12.4.5.3 Tabellenkopf-Erläuterung zu den Objektbeschreibungen

Tabelle 192: Erläuterung der Tabellenköpfe in den Objektbeschreibungen

Spaltenüberschrift	Beschreibung
Attribut ID	Integerwert, der dem entsprechenden Attribut zugeordnet ist
Zugriff	**Set:** Auf das Attribut kann mittels des Dienstes Set_Attribute zugegriffen werden (Schreiben/Verändern des Attribut-Wertes). **Hinweis** **Ansprechen auch mit Get_Attribute-Dienst möglich!** Unterstützt ein Attribut den Dienst Set_Attribute, so kann dieses auch mit dem Dienst Get_Attribute angesprochen werden. **Get:** Auf das Attribut kann mittels Get_Attribute- Services zugegriffen werden (Lesen des Attribut-Wertes). **Get_Attribute_All:** Liefert den Inhalt aller Attribute. **Set_Attribute_Single:** Modifiziert einen Attribut-Wert. **Reset:** Führt einen Neustart durch. 0: Neustart 1: Neustart und Wiederherstellen der Werkseinstellungen
NV	**NV (non volatile):** Das Attribut wird permanent im Feldbuskoppler/-controller gespeichert.
	V (volatile): Das Attribut wird nicht permanent im Feldbuskoppler/-controller gespeichert. **Hinweis** **Ohne Angabe wird das Attribut nicht gespeichert!** Ist diese Spalte nicht vorhanden, sind alle Attribute vom Typ V (volatile).
Name	Bezeichnung des Attributs.
Datentyp	Bezeichnung des CIP-Datentyps des Attributes
Beschreibung	Kurze Beschreibung zu dem Attribut.
Defaultwert	Werkseinstellung.

12.4.5.4 Identity (01 $_{hex}$)

Die Klasse „Identity“ dient dazu, allgemeine Informationen des Feldbuskopplers/ -controllers bereitzustellen, die diesen eindeutig identifizieren.

Instanz 0 (Klassenattribute)

Tabelle 193: Identity (01 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Maximale Instanz	1 (0x0001)
3	Get	Max ID Number of Class Attributes	UINT	Maximale Anzahl der Klassen-Attribute (nur mit dem Dienst Get_Attribute_All)	0 (0x0000)
4	Get	Max ID Number of Instance Attribute	UINT	Maximale Anzahl der Instanz-Attribute (nur mit dem Dienst Get_Attribute_All)	0 (0x0000)

Instanz 1

Tabelle 194: Identity (01 $_{hex}$) – Instanz 1

Attribut ID	Zu-griff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Vendor ID	UINT	Hersteller-identifikation	40 (0x0028)
2	Get	Device Type	UINT	Generelle Typ-bezeichnung des Produktes	12 (0x000C)
3	Get	Product Code	UINT	Bezeichnung des Feldbus-kopplers/ -controllers	750-841 (in hex)
4	Get	Revision	STRUCT of:	Revision des Identity-Objektes	Firmware abhängig
		Major Revision	UINT		
		Minor Revision	UINT		
5	Get	Status	WORD	Aktueller Status des Gerätes	Bit 0 – Zuweisung zu einem Master Bit 1 = 0 – reserviert Bit 2 – (konfiguriert) = 0 – Konfiguration ist unverändert = 1 – Konfiguration weicht von Hersteller-parametern ab Bit 3 = 0 – reserviert Bit 4-7 – Extended Device Status =0010 – mind. eine fehlerhafte I/O-Verbindung =0011 – keine I/O-Verbindung hergestellt Bit 8-11 – nicht genutzt Bit 12-15 =0 – reserviert
6	Get	Serial Number	UDINT	Seriennummer	die letzten 4 Stellen der MAC-ID
7	Get	Product Name	SHORT_ STRING	Produktname	Programmierbarer Feldbuscontroller ETHERNET TCP/IP 750-841



Common Services

Tabelle 195: Identity (01 $_{hex}$) – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 $_{hex}$	Ja	Ja	Get_Attribute_All	Liefert den Inhalt aller Attribute
05 $_{hex}$	Nein	Ja	Reset	Führt einen Neustart durch. 0: Neustart 1: Neustart und Wiederherstellen der Werkseinstellungen
0E $_{hex}$	Nein	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.5 Message Router (02 $_{hex}$)

Das „Message Router Object“ stellt Verbindungspunkte in Form von Klassen oder Instanzen bereit, welche einen Client zum Adressieren von Diensten (Lesen, Schreiben) nutzen kann. Diese Nachrichten können sowohl verbindungsorientiert (connected) als auch verbindungslos (unconnected) vom Client an den Feldbuskoppler gesendet werden.

Instanz 0 (Klassenattribute)

Tabelle 196: Message Router (02 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	1 (0x0001)
2	Get	Number of Attributes	UINT	Anzahl der Attribute	0 (0x0000)
3	Get	Number of Services	UINT	Anzahl der Dienste	0 (0x0000)
4	Get	Max ID Number of Class Attributes	UINT	Maximale Anzahl der Klassen-Attribute	0 (0x0000)
5	Get	Max ID Number of Instance Attributes	UINT	Maximale Anzahl der Instanz-Attribute	0 (0x0000)

Hinweis



Nur Dienst Get_Attribute_All anwendbar!
Die Klassen-Attribute sind nur mit dem Dienst Get_Attribute_All erreichbar.

Instanz 1

Tabelle 197: Message Router (02 $_{hex}$) – Instanz 1

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	ObjectList	STRUCT of:	-	
		Number	UINT	Anzahl implementierter Klassen	40 (0x0028)
		Classes	UINT	Implementierte Klassen	01 00 02 00 04 00 06 00 F4 00 F5 00 F6 00 64 00 65 0066 0067 00 68 00 69 00 6A 00 6B 00 6C 00 6D 00 6E 00 6F 00 70 00 71 00 72 00 73 00 74 00 80 00 81 00 A0 00 A1 00 A2 00 A6 00 A7 00 AA 00 AB 00 A3 00 A4 00 A5 00 A8 00 A9 00 AC 00 AD 00
2	Get	NumberAvailable	UINT	Maximale Anzahl von unterschiedlichen Verbindungen	128 (0x0080)

Common Services

Tabelle 198: Message Router (02 $_{hex}$) – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 $_{hex}$	Ja	Nein	Get_Attribute_All	Liefert den Inhalt aller Attribute
0E $_{hex}$	Nein	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.6 Assembly (04 $_{hex}$)

Mit Hilfe der „Assembly"-Klasse lassen sich mehrere auch verschiedenartige Objekte zusammenfassen. Diese können z. B. Ein- und Ausgangsdaten, Status- und Steuerinformationen oder Diagnoseinformationen sein. WAGO nutzt hier die herstellerspezifischen Instanzen, um diese Objekte in verschiedenen Anordnungen für Sie bereitzustellen. Hierdurch steht Ihnen ein effizienter Weg zum Austausch von Prozessdaten zur Verfügung. Im Folgenden werden die einzelnen statischen Assembly Instanzen mit deren Inhalten und Anordnungen beschrieben.

Instanz 0 (Klassenattribute)

Tabelle 199: Assembly (04 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	2 (0x0002)
2	Get	Max Instance	UINT	Höchste Instanz	111 (0x006F)



Übersicht der Instanzen

Tabelle 200: Statische Assembly-Instanzen – Übersicht

Instanz	Beschreibung
Instanz 101 (65_{hex})	für analoge und digitale Ausgangsdaten sowie Feldbus-Eingangsvariablen
Instanz 102 (66_{hex})	für digitale Ausgangsdaten und Feldbus-Eingangsvariablen
Instanz 103 (67_{hex})	für analoge Ausgangsdaten und Feldbus-Eingangsvariablen
Instanz 104 (68_{hex})	für analoge und digitale Eingangsdaten, Status und Feldbus-Ausgangsvariablen
Instanz 105 (69_{hex})	für digitale Eingangsdaten, Status und Feldbus-Ausgangsvariablen
Instanz 106 ($6A_{hex}$)	für analoge Eingangsdaten, Status und Feldbus-Ausgangsvariablen
Instanz 107 ($6B_{hex}$)	für digitale und analoge Eingangsdaten und Feldbus-Ausgangsvariablen
Instanz 108 ($6C_{hex}$)	für digitale Eingangsdaten und Feldbus-Ausgangsvariablen
Instanz 109 ($6D_{hex}$)	für analoge Eingangsdaten und Feldbus-Ausgangsvariablen
Instanz 110 ($6E_{hex}$)	für Feldbus-Ausgangsvariablen
Instanz 111 ($6F_{hex}$)	für Feldbus-Eingangsvariablen

Instanz 101 (65_{hex})

Diese Assembly-Instanz enthält analoge und digitale Ausgangsdaten. Eventuell definierte Feldbus-Eingangsvariablen werden hinter diesen angehängt.

Tabelle 201: Statische Assembly-Instanzen – Instanz 101 (65_{hex})

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get/Set	Data	ARRAY of BYTE	Es sind analoge und digitale Ausgangsdaten sowie eventuell Feldbus-Eingangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UNIT	Anzahl der Bytes im Prozessabbild	-

Instanz 102 (66_{hex})

Diese Assembly-Instanz enthält nur digitale Ausgangsdaten und Feldbus-Eingangsvariablen.

Tabelle 202: Statische Assembly-Instanzen – Instanz 102 (66_{hex})

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get/Set	Data	ARRAY of BYTE	Es sind digitale Ausgangsdaten und Feldbus-Eingangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 103 (67_{hex})

Diese Assembly-Instanz enthält nur analoge Ausgangsdaten und Feldbus-Eingangsvariablen.

Tabelle 203: Statische Assembly-Instanzen – Instanz 103 (67 hex)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get/Set	Data	ARRAY of BYTE	Es sind analoge Ausgangsdaten und Feldbus-Eingangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 104 (68 hex)

Diese Assembly-Instanz enthält analoge und digitale Eingangsdaten, Status (= Wert aus Klasse 100, Instanz 1, Attribut 5) und Feldbus-Ausgangsvariablen.

Tabelle 204: Statische Assembly-Instanzen – Instanz 104 (68 hex)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind analoge und digitale Eingangsdaten, Status und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 105 (69 hex)

Diese Assembly-Instanz enthält nur digitale Eingangsdaten, Status (= Wert aus Klasse 100, Instanz 1, Attribut 5) und Feldbus-Ausgangsvariablen.

Tabelle 205: Statische Assembly-Instanzen – Instanz 105 (69 hex)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind digitale Eingangsdaten, Status und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 106 (6A hex)

Diese Assembly-Instanz enthält nur analoge Eingangsdaten, Status (= Wert aus Klasse 100, Instanz 1, Attribut 5) und Feldbus-Ausgangsvariablen.

Tabelle 206: Statische Assembly-Instanzen – Instanz 106 (6A hex)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind analoge Eingangsdaten, Status und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-



Instanz 107 (6B $_{hex}$)

Diese Assembly-Instanz enthält analoge und digitale Eingangsdaten und Feldbus-Ausgangsvariablen.

Tabelle 207: Statische Assembly-Instanzen – Instanz 107 (6B $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind analoge und digitale Eingangsdaten und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 108 (6C $_{hex}$)

Diese Assembly-Instanz enthält nur digitale Eingangsdaten und Feldbus-Ausgangsvariablen.

Tabelle 208: Statische Assembly-Instanzen – Instanz 108 (6C $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind digitale Eingangsdaten und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 109 (6D $_{hex}$)

Diese Assembly-Instanz enthält nur analoge Eingangsdaten und Feldbus-Ausgangsvariablen.

Tabelle 209: Statische Assembly-Instanzen – Instanz 109 (6D $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind analoge Eingangsdaten und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 110 (6E $_{hex}$)

Tabelle 210: Statische Assembly Instanzen – Instanz 110 (6E $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Referenz auf das Prozessabbild: nur PFC-Ausgangsvariable	-

Instanz 111 (6F $_{hex}$)

Tabelle 211: Statische Assembly Instanzen – Instanz 111 (6F $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Set	Data	ARRAY of BYTE	Referenz auf das Prozessabbild: nur PFC-Eingangsvariable	-

Instanz 198 (C6 $_{hex}$) „Input Only“

Mit dieser Instanz wird eine Verbindung aufgebaut, wenn keine Ausgänge angesprochen werden sollen bzw. wenn Eingänge abgefragt werden, die schon in einer Exclusive-Owner-Verbindung benutzt werden. Die Datenlänge dieser Instanz beträgt immer Null.
Diese Instanz kann nur im „Consumed Path” (aus Sicht des Slave-Gerätes) benutzt werden.

Instanz 199 (C7 $_{hex}$) „Listen Only“

Mit dieser Instanz wird eine Verbindung aufgebaut, die auf einer vorhandenen Exclusive-Owner-Verbindung aufsetzt. Dabei hat die neue Verbindung die gleichen Übertragungsparameter, wie die Exclusive-Owner-Verbindung. Wird die Exclusive-Owner-Verbindung abgebaut, wird auch automatisch diese Verbindung abgebaut. Die Datenlänge dieser Instanz beträgt immer Null.
Diese Instanz kann nur im „Consumed Path” (aus Sicht des Slave-Gerätes) benutzt werden.

Common Services

Tabelle 212: Statische Assembly-Instanzen – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

Das Schreiben des Attributes 3 der Assembly-Instanzen 101, 102 und 103 wird von der Software überprüft. Die Überschreitung von Grenzwerten wird festgestellt und, sofern erforderlich, korrigiert. Es wird jedoch keine Schreibanfrage abgelehnt. Das bedeutet, wenn weniger Daten empfangen werden als erwartet, dann werden nur diese Daten geschrieben. Wenn mehr Daten empfangen werden als erwartet, dann werden die empfangenen Daten an der oberen Grenze entfernt. Jedoch wird im Falle von expliziten Nachrichten ein definiertes CIP generiert, obwohl die Daten geschrieben worden sind.



12.4.5.7 Port Class Object (F4 hex)

Das „Port Class Object“ spezifiziert die an dem Feldbuskoppler/-controller vorhandenen CIP-Ports. Für jeden CIP-Port gibt es eine Instanz.

Instanz 0 (Klassenattribute)

Tabelle 213: Port Class (F4 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	1 (0x0001)
3	Get	Num Instances	UINT	Anzahl von aktuellen Ports	1 (0x0001)
8	Get	Entry Port	UINT	Instanz des Portobjektes, von wo die Anfrage eingetroffen ist	1 (0x0001)
9	Get	All Ports	Array of Struct UINT UINT	Array von Instanz-Attributen 1 und 2 aller Instanzen	0 (0x0000) 0 (0x0000) 4 (0x0004) 2 (0x0002)

Instanz 1

Tabelle 214: Port Class (F4 hex) – Instanz 1

Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Defaultwert
1	Get	V	Port Type	UINT	-	4 (0x0004)
2	Get	V	Port Number	UINT	CIP Portnummer	2 (0x0002) (EtherNet/IP)
3	Get	V	Port Object	UINT	Anzahl von 16 Bit Wörtern im folgenden Pfad	2 (0x0002)
				Padded EPATH	Objekt, das diesen Port verwaltet	0x20 0xF5 0x24 0x01 (entspricht dem TCP/IP-Interface Object)
4	Get	V	Port Name	SHORT_ STRING	Portname	„“
7	Get	V	Node Address	Padded EPATH	Portsegment (IP-Adresse)	Abhängig von der IP-Adresse

Common Services

Tabelle 215: Port Class (F4 hex) – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 hex	Ja	Ja	Get_Attribute_All	Liefert den Inhalt aller Attribute
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.8 TCP/IP Interface Object (F5 $_{hex}$)

Das „TCP/IP Interface Object“ stellt die Einrichtung zur Konfiguration der TCP-IP-Netzwerk-Schnittstelle eines Feldbuskopplers/-controllers bereit.
Beispiele konfigurierbarer Objekte umfassen die IP-Adresse, Netzwerkmaske und Gateway-Adresse des Feldbuskopplers/-controllers.

Bei der zugrunde liegenden physikalischen Kommunikationsschnittstelle, die mit dem TCP/IP-Schnittstellen-Objekt verbunden ist, kann es sich um eine beliebige Schnittstelle handeln, die das TCP/IP-Protokoll unterstützt.
An einem TCP/IP-Schnittstellen-Objekt kann zum Beispiel eine der folgenden Komponenten angeschlossen werden: eine Ethernet-Schnittstelle 802.3, eine ATM-Schnittstelle (Asynchronous Transfer Mode-Schnittstelle) oder eine serielle Schnittstelle für Protokolle wie PPP (Point-to-Point Protocol).
Das TCP/IP-Schnittstellen-Objekt stellt ein Attribut bereit, welches das linkspezifische Objekt für die angeschlossene physikalische Kommunikationsschnittstelle identifiziert. Das linkspezifische Objekt soll üblicherweise linkspezifische Zähler sowie beliebige linkspezifische Konfigurationsattribute bereitstellen.

Jedes Gerät muss genau eine Instanz des TCP/IP-Schnittstellen-Objektes für jede TCP/IP-fähige Kommunikationsschnittstelle unterstützen. Eine Anfrage für den Zugriff auf die 1. Instanz des TCP/IP-Schnittstellen-Objektes muss sich immer auf die Instanz beziehen, die mit der Schnittstelle verbunden ist, über welche die Anfrage eingegangen ist.

Instanz 0 (Klassenattribute)

Tabelle 216: TCP/IP Interface (F5 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	1 (0x0001)
3	Get	Num Instances	UINT	Anzahl der aktuell instanzierten Verbindungen	1 (0x0001)



Instanz 1

Tabelle 217: TCP/IP Interface (F5 $_{hex}$) – Instanz 1

Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Defaultwert
1	Get	V	Status	DWORD	Interface-Status	-
2	Get	V	Configuration Capability	DWORD	Inferface-Flags für mögliche Konfigurationsarten	0x00000017
3	Set	NV	Configuration Control	DWORD	Legt fest wie der Feldbuskoppler/-controller nach dem ersten Neustart zu seiner TCP/IP Konfiguration kommt	0x00000011
4	Get	V	Physical Link Object	STRUCT of		
			Path size	UINT	Anzahl von 16-Bit-Wörtern im folgenden Pfad	0x0002
			Path	Padded EPATH	Logischer Pfad, der auf das physikalische Link-Objekt zeigt	0x20 0xF6 0x24 0x03 (entspricht dem Ethernet Link Object)
5	Set	NV	Interface Configuration	STRUCT of	-	
			IP Address	UDINT	IP-Adresse	0
			Network Mask	UDINT	Netzwerkmaske	0
			Gateway Address	UDINT	IP-Adresse des Standard-Gateway	0
			Name Server	UDINT	IP-Adresse des primären Name-Servers	0
			Name Server 2	UDINT	IP-Adresse des sekundären Name-Servers	0
			Domain Name	STRING	Default-Domain-Name	„“
6	Set	NV	Host Name	STRING	Gerätename	„“

Common Services

Tabelle 218: TCP/IP Interface (F5 $_{hex}$) – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 $_{hex}$	Ja	Ja	Get_Attribute_All	Liefert den Inhalt aller Attribute
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.4.5.9 Ethernet Link Object (F6 $_{hex}$)

Das „Ethernet Link Object“ enthält linkspezifische Zähler- und Statusinformationen für eine Kommunikationsschnittstelle vom Typ Ethernet 802.3. Jedes Gerät muss genau eine Instanz des Ethernet-Link-Objektes für jede Kommunikationsschnittstelle vom Typ Ethernet IEEE 802.3 unterstützen. Für die Geräte kann auch eine Ethernet-Link-Objektinstanz für eine interne Schnittstelle verwendet werden, wie zum Beispiel ein interner Port mit integriertem Switch.

Instanz 0 (Klassenattribute)

Tabelle 219: Ethernet Link (F6 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	3 (0x0003)
2	Get	Max Instance	UDINT	Max. Anzahl von Instanzen	3 (0x0003)
3	Get	Num Instances	UDINT	Anzahl der aktuell instanzierten Verbindungen	3 (0x0003)

Instanz 1

Tabelle 220: Ethernet Link (F6 $_{hex}$) – Instanz 1

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Interface Speed	UDINT	Übertragungs-geschwindigkeit	10 (0x0A) oder 100 (0x64)
2	Get	Interface Flags	DWORD	Interface Konfigurations-/ Statusinformationen	Bit 0: Link aktiv Bit 1: Full Duplex
3	Get	Physical Address	ARRAY of 6 UINTs	MAC layer address	MAC-ID des Gerätes

Common Service

Tabelle 221: Ethernet link (F6 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 $_{hex}$	Ja	Ja	Get_Attribute_All	Liefert den Inhalt aller Attribute
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes



12.4.5.10 Coupler/Controller Configuration Object (64 hex)

Die Konfigurationsklasse des Feldbuskopplers/-controllers ermöglicht das Lesen und Konfigurieren einiger wichtiger Prozessparameter des Feldbusses. Die folgende Auflistung erklärt ausführlich alle unterstützten Instanzen und Attribute.

Instanz 0 (Klassenattribute)

Tabelle 222: Coupler/Controller Configuration (64 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	1 (0x0001)

Instanz 1

Tabelle 223: Coupler/Controller Configuration (64 hex) – Instanz 1

Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Default-wert
5 (0x05)	Get	V	ProcessState	USINT	Koppler-/Controller-Status, Fehlermaske: Bit 0: Klemmenbusfehler Bit 3: Klemmendiagnose (0x08) Bit 7: Feldbusfehler (0x80)	0
6 (0x06)	Get	V	DNS_i_ Trmnldia	UINT	Klemmendiagnose: Bit 0..7: Klemmennummer Bit 8..14: Klemmenkanal Bit 15: 0/1 Fehler behoben/aufgetreten	0
7 (0x07)	Get	V	CnfLen. AnalogOut	UINT	Anzahl I/O-Bits für die analogen Ausgänge	-
8 (0x08)	Get	V	CnfLen. AnalogInp	UINT	Anzahl I/O-Bits für die analogen Eingänge	-
9 (0x09)	Get	V	CnfLen. DigitalOut	UINT	Anzahl I/O-Bits für die digitalen Ausgänge	-
10 (0x0A)	Get	V	CnfLen. DigitalInp	UINT	Anzahl I/O-Bits für die digitalen Eingänge	-
11 (0x0B)	Set	NV	Bk_Fault_ Reaction	USINT	Feldbusfehlerreaktion 0: stoppt lokale I/O-Zyklen 1: alle Ausgänge zu 0 setzen 2: keine Fehlerreaktion 3: keine Fehlerreaktion 4: PFC-Task übernimmt die Kontrolle der Ausgänge (gilt für Controller)	1
12..26 (0x0C...0x1A)	Reserviert aus Kompatibilität zu DeviceNet					
40..43 (0x28...0x2B)	Reserviert aus Kompatibilität zu DeviceNet					
45 (0x2D)	Get	V	Bk_Led_Err_ Code	UINT	I/O-LED Error-Code	0
46 (0x2E)	Get	V	Bk_Led_Err_ Arg	UINT	I/O-LED Error-Argument	0

Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Default-wert
47 (0x2F)	Get	V	Bk_Diag_Value	UINT	Enthält das Diagnosebyte. **Achtung:** Dieses Attribut muss vor dem Attribut 6 (DNS_i_Trmnldia) gelesen werden, da mit dem Lesen von Attribut 6 das Diagnose-byte von der nächsten Diagnose anliegt.	0
100 (0x64)	Set	NV	Bk_FbInp_Var_Cnt	UINT	Bestimmt die Anzahl der Bytes für die PFC-Eingangs-variablen, die zu dem Assembly-Objekt hinzugefügt werden. Diese Anzahl wird zu dem konsumierenden Pfad hinzugezählt. Assembly-Instanzen 101...103	0
101 (0x65)	Set	NV	Bk_FbOut_Var_Cnt	UINT	Bestimmt die Anzahl der Bytes für die PFC-Ausgangs-variablen, die zu dem Assembly-Objekt hinzugefügt werden. Diese Anzahl wird zu dem produzierenden Pfad hinzugezählt. Assembly-Instanzen 104...109	0
102 (0x66)	Set	NV	Bk_FbInp_Plc Only_Var_Cnt	UINT	Bestimmt die Anzahl der Bytes für die PFC-Eingangs-variablen, die mittels Assembly-Instanz 111 empfangen werden	4
103 (0x67)	Set	NV	Bk_FbInp_Start Plc_Var_Cnt	UINT	Legt fest, ab welcher Position die PFC-Eingangsvariablen für die Assembly-Instanz 111 empfangen werden	0
104 (0x68)	Set	NV	Bk_FbOut_Plc Only_Var_Cnt	UINT	Bestimmt die Anzahl der Bytes für die PFC-Ausgangs-variablen die mittels Assembly-Instanz 110 übertragen werden	4
105 (0x69)	Set	NV	Bk_FbOut_Start Plc_Var_Cnt	UINT	Legt fest, ab welcher Position die PFC-Ausgangsvariablen für die Assembly-Instanz 110 übertragen werden	0



Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Default-wert
120 (0x78)	Set	NV	Bk_Header CfgOT	UINT	Gibt an, ob der RUN/IDLE-Header benutzt wird Originator → Target Richtung 0: wird verwendet 1: wird nicht verwendet	0x0000
121(0x79)	Set	NV	Bk_Header CfgTO	UINT	Gibt an, ob der RUN/IDLE-Header benutzt wird Target → Originator Richtung 0: wird verwendet 1: wird nicht verwendet	0x0001

Common Service

Tabelle 224: Coupler/Controller Configuration (64 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.4.5.11 Discrete Input Point (65 $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten eines bestimmten digitalen Eingangspunktes.

Instanz 0 (Klassenattribute)

Tabelle 225: Discrete Input Point (65 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1 ... 255 (1. bis 255. digitaler Eingangswert)

Tabelle 226: Discrete Input Point (65 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DipObj_Value	BYTE	Digitaler Eingang (nur Bit 0 gültig)	-

Common Services

Tabelle 227: Discrete Input Point (65 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.12 Discrete Input Point Extended 1 (69 hex)

Die Erweiterung der Klasse „Discrete Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 255 digitale Eingangspunkte (DIPs) enthält. Der Instanzbereich der Klasse “Discrete Input Point Extended 1” deckt die DIPs von 256 bis 510 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 228: Discrete Input Point Extended 1(69 hex,) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 ... 510 (256. bis 510. digitaler Eingangswert)

Tabelle 229: Discrete Input Point Extended 1 (69 hex,) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DipObj_Value	BYTE	Digitaler Eingang (nur Bit 0 gültig)	-

Common Services

Tabelle 230: Discrete Input Point Extended 1 (69 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.13 Discrete Input Point Extended 2 (6D hex)

Die Erweiterung der Klasse „Discrete Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 510 digitale Eingangspunkte (DIPs) enthält. Der Instanzbereich der Klasse “Discrete Input Point Extended 2” deckt die DIPs von 511 bis 765 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 231: Discrete Input Point Extended 2 (6D hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 511 ... 765 (511. bis 765. digitaler Eingangswert)



Tabelle 232: Discrete Input Point Extended 2 (6D $_{hex}$) – Instanz 511...765

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DipObj_Value	BYTE	Digitaler Eingang (nur Bit 0 gültig)	-

Common Services

Tabelle 233: Discrete Input Point Extended 2 (6D $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.14 Discrete Input Point Extended 3 (71 $_{hex}$)

Die Erweiterung der Klasse „Discrete Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 765 digitale Eingangspunkte (DIPs) enthält. Der Instanzbereich der Klasse “Discrete Input Point Extended 3” deckt die DIPs von 766 bis 1020 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 234: Discrete Input Point Extended 3 (71 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 766 ... 1020 (766. bis 1020. digitaler Eingangswert)

Tabelle 235: Discrete Input Point Extended 3 (71 $_{hex}$) – Instanz 766...1020

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DipObj_Value	BYTE	Digitaler Eingang (nur Bit 0 gültig)	-

Common Services

Tabelle 236: Discrete Input Point Extended 3 (71 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes



12.4.5.15 Discrete Output Point (66 hex)

Diese Klasse ermöglicht den Austausch von Daten für einen bestimmten digitalen Ausgangspunkt.

Instanz 0 (Klassenattribute)

Tabelle 237: Discrete Output Point (66 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1 ... 255 (1. bis 255. digitaler Ausgangswert)

Tabelle 238: Discrete Output Point (66 hex) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DopObj_Value	BYTE	Digitaler Ausgang (nur Bit 0 gültig)	-

Common Services

Tabelle 239: Discrete Output Point (66 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 hex	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.4.5.16 Discrete Output Point Extended 1 (6A $_{hex}$)

Die Erweiterung der Klasse „Discrete Output Point“ ermöglicht den Austausch von Daten eines Feldbusknotens, der über 255 digitale Ausgangspunkte (DOPs) enthält. Der Instanzbereich der Klasse “Discrete Output Point Extended 1” deckt die DOPs von 256 bis 510 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 240: Discrete Output Point Extended 1 (6A $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 ... 510 (256. bis 510. digitaler Ausgangswert)

Tabelle 241: Discrete Output Point Extended 1 (6A $_{hex}$) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DopObj_Value	BYTE	Digitaler Ausgang (nur Bit 0 gültig)	-

Common Services

Tabelle 242: Discrete Output Point Extended 1 (6A $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert



12.4.5.17 Discrete Output Point Extended 2 (6E $_{hex}$)

Die Erweiterung der Klasse „Discrete Output Point" ermöglicht den Austausch von Daten eines Feldbusknotens, der über 510 digitale Ausgangspunkte (DOPs) enthält. Der Instanzbereich der Klasse "Discrete Output Point Extended 2" deckt die DOPs von 511 bis 765 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 243: Discrete Output Point Extended 2 (6E $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 511 ... 765 (511. bis 765. digitaler Ausgangswert)

Tabelle 244: Discrete Output Point Extended 2 (6E $_{hex}$) – Instanz 511...765

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DopObj_Value	BYTE	Digitaler Ausgang (nur Bit 0 gültig)	-

Common Services

Tabelle 245: Discrete Output Point Extended 2 (6E $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.4.5.18 Discrete Output Point Extended 3 (72 $_{hex}$)

Die Erweiterung der Klasse „Discrete Output Point“ ermöglicht den Austausch von Daten eines Feldbusknotens, der über 765 digitale Ausgangspunkte (DOPs) enthält. Der Instanzbereich der Klasse “Discrete Output Point Extended 2” deckt die DOPs von 766 bis 1020 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 246: Discrete Output Point Extended 3 (72 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 766 ... 1020 (766. bis 1020. digitaler Ausgangswert)

Tabelle 247: Discrete Output Point Extended 3 (72 $_{hex}$) – Instanz 766...1020

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DopObj_Value	BYTE	Digitaler Ausgang (nur Bit 0 gültig)	-

Common Services

Tabelle 248: Discrete Output Point Extended 2 (6E $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert



12.4.5.19 Analog Input Point (67 $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten eines bestimmten analogen Eingangspunktes (AIP). Ein analoger Eingangspunkt ist ein Teil eines analogen Eingangsmoduls.

Instanz 0 (Klassenattribute)

Tabelle 249: Analog Input Point (67 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1 ... 255 (1. bis 255. analoger Eingangswert)

Tabelle 250: Analog Input Point (67 $_{hex}$) – Instanz 1 ... 255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AipObj_Value	ARRAY of BYTE	Analoger Eingang	-
2	Get	AipObj_Value_ Length	USINT	Länge der Eingangsdaten AipObj_Value (in Byte)	-

Common Services

Tabelle 251: Analog Input Point (67 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.20 Analog Input Point Extended 1 (6B hex)

Die Erweiterung der Klasse „Analog Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 255 analoge Ausgänge (AIPs) enthält. Der Instanzbereich der Klasse “Analog Input Point Extended 1” deckt die AIPs von 256 bis 510 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 252: Analog Input Point Extended 1 (6B hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 ... 510 (256. bis 510. analoger Eingangswert)

Tabelle 253: Analog Input Point Extended 1 (6B hex) – Instanz 256 ... 510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AipObj_Value	ARRAY of BYTE	Analoger Eingang	-
2	Get	AipObj_Value_ Length	USINT	Länge der Eingangsdaten AipObj_Value (in Byte)	-

Common Services

Tabelle 254: Analog Input Point Extended 1 (6B hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes



12.4.5.21 Analog Input Point Extended 2 (6F $_{hex}$)

Die Erweiterung der Klasse „Analog Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 510 analoge Ausgänge (AIPs) enthält. Der Instanzbereich der Klasse “Analog Input Point Extended 2” deckt die AIPs von 511 bis 765 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 255: Analog Input Point Extended 2 (6F $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 511 ... 765 (511. bis 765. analoger Eingangswert)

Tabelle 256: Analog Input Point Extended 2 (6F $_{hex}$) – Instanz 511 ... 765

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AipObj_Value	ARRAY of BYTE	Analoger Eingang	-
2	Get	AipObj_Value_ Length	USINT	Länge der Eingangsdaten AipObj_Value (in Byte)	-

Common Services

Tabelle 257: Analog Input Point Extended 2 (6F $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.22 Analog Input Point Extended 3 (73 $_{hex}$)

Die Erweiterung der Klasse „Analog Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 765 analoge Ausgänge (AIPs) enthält. Der Instanzbereich der Klasse “Analog Input Point Extended 3” deckt die AIPs von 766 bis 1020 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 258: Analog Input Point Extended 3 (73 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 766 ... 1020 (766. bis 1020. analoger Eingangswert)

Tabelle 259: Analog Input Point Extended 3 (73 $_{hex}$) – Instanz 766 ... 1020

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AipObj_Value	ARRAY of BYTE	Analoger Eingang	-
2	Get	AipObj_Value_ Length	USINT	Länge der Eingangsdaten AipObj_Value (in Byte)	-

Common Services

Tabelle 260: Analog Input Point Extended 3 (73 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes



12.4.5.23 Analog Output Point (68 $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten eines bestimmten analogen Ausgangspunktes (AOP). Ein analoger Ausgangspunkt ist ein Teil eines analogen Ausgangsmoduls.

Instanz 0 (Klassenattribute)

Tabelle 261: Analog Output Point (68 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1 ... 255 (1. bis 255. Analoger Ausgangswert)

Tabelle 262: Analog Output Point (68 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AopObj_Value	ARRAY of BYTE	Analoger Ausgang	-
2	Get	AopObj_Value _Length	USINT	Länge der Ausgangsdaten AopObj_Value (in Byte)	-

Common Services

Tabelle 263: Analog Output Point (68 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.4.5.24 Analog Output Point Extended 1 (6C hex)

Die Erweiterung der Klasse „Analog Output Point“ ermöglicht den Austausch von Daten eines Feldbusknotens, der über 255 analoge Ausgangspunkte (AOPs) enthält. Der Instanzbereich der Klasse “Discrete Output Point Extended 1” deckt die AOPs von 256 bis 510 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 264: Analog Output Point Extended 1 (6C hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 ... 510 (256. bis 510. Analoger Ausgangswert)

Tabelle 265: Analog Output Point Extended 1 (6C hex) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AopObj_Value	ARRAY of BYTE	Analoger Ausgang	-
2	Get	AopObj_Value _Length	USINT	Länge der Ausgangsdaten AopObj_Value (in Byte)	-

Common Services

Tabelle 266: Analog Output Point Extended 1 (6C hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 hex	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert



12.4.5.25 Analog Output Point Extended 2 (70 $_{hex}$)

Die Erweiterung der Klasse „Analog Output Point" ermöglicht den Austausch von Daten eines Feldbusknotens, der über 510 analoge Ausgangspunkte (AOPs) enthält. Der Instanzbereich der Klasse "Discrete Output Point Extended 2" deckt die AOPs von 511 bis 765 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 267: Analog Output Point Extended 2 (70 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 511 ... 765 (511. bis 765. Analoger Ausgangswert)

Tabelle 268: Analog Output Point Extended 2 (70 $_{hex}$) – Instanz 511...765

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AopObj_Value	ARRAY of BYTE	Analoger Ausgang	-
2	Get	AopObj_Value _Length	USINT	Länge der Ausgangsdaten AopObj_Value (in Byte)	-

Common Services

Tabelle 269: Analog Output Point Extended 2 (70 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.4.5.26 Analog Output Point Extended 3 (74 $_{hex}$)

Die Erweiterung der Klasse „Analog Output Point“ ermöglicht den Austausch von Daten eines Feldbusknotens, der über 765 analoge Ausgangspunkte (AOPs) enthält. Der Instanzbereich der Klasse “Discrete Output Point Extended 3” deckt die AOPs von 766 bis 1020 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 270: Analog Output Point Extended 3 (74 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 766 ... 1020 (766. bis 1020. Analoger Ausgangswert)

Tabelle 271: Analog Output Point Extended 3 (74 $_{hex}$) – Instanz 766...1020

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AopObj_Value	ARRAY of BYTE	Analoger Ausgang	-
2	Get	AopObj_Value _Length	USINT	Länge der Ausgangsdaten AopObj_Value (in Byte)	-

Common Services

Tabelle 272: Analog Output Point Extended 3 (74 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert



12.4.5.27 Module Configuration (80 $_{hex}$)

Instanz 0 (Klassenattribute)

Tabelle 273: Module Configuration (80 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1..255 (0. bis 254. Klemme)

Tabelle 274: Module Configuration (80 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Default-wert
1	Get	ModulDescription	WORD	Beschreibung angeschlossener Klemmen (Klemme 0 = Koppler/ Controller) Bit 0: Klemme hat Eingänge Bit 1: Klemme hat Ausgänge Bit 8-14: Datenbreite intern in Bit Bit 15: 0/1 Analoge/Digitale Klemme Bei analogen Klemmen bezeich-nen die Bits 0-14 den Klemmentyp, z. B. 401 für die Klemme 750-401	-

Common Services

Tabelle 275: Module Configuration (80 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.28 Module Configuration Extended (81 $_{hex}$)

Wie „Module Configuration (80 $_{hex}$)“, jedoch enthält diese Klasse nur die Beschreibung von Klemme 255.

Instanz 0 (Klassenattribute)

Tabelle 276: Module Configuration Extended (81 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 (255. Klemme)

Tabelle 277: Module Configuration Extended (81 $_{hex}$) – Instanz 256

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Default-wert
1	Get	ModulDescription	WORD	Beschreibung angeschlossener Klemmen (Klemme 0 = Koppler/ Controller) Bit 0: Klemme hat Eingänge Bit 1: Klemme hat Ausgänge Bit 8-14: Datenbreite intern in Bit Bit 15: 0/1 Analoge/Digitale Klemme Bei analogen Klemmen bezeichnen die Bits 0-14 den Klemmentyp, z. B. 401 für die Klemme 750-401	-

Common Services

Tabelle 278: Module Configuration Extended (81 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.29 Input Fieldbus Variable USINT (A0 $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten einer bestimmten SPS-Eingangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Eingangsvariablen %IB2552...%IB2806.



Instanz 0 (Klassenattribute)

Tabelle 279: Input Fieldbus Variable USINT (A0 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 1...255 (1. bis 255. Eingangsvariable)

Tabelle 280: Input Fieldbus Variable USINT (A0 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	USINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 281: Input fieldbus variable USINT (A0 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.4.5.30 Input Fieldbus Variable USINT Extended 1 (A1 $_{hex}$)

Die Erweiterung der Klasse „Input Fieldbus Variable USINT“ ermöglicht das Lesen von SPS-Eingangsvariablendaten. Der Instanzbereich der Klasse “Input Fieldbus Variable USINT Extended 1“ deckt die SPS-Eingangsvariablendaten von 256 bis 510 ab.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Eingangsvariablen %IB2807...%IB3061.

Instanz 0 (Klassenattribute)

Tabelle 282: Input Fieldbus Variable USINT Extended 1 (A1 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 256...510 (256. bis 510. Eingangsvariable)

Tabelle 283: Input Fieldbus Variable USINT Extended 1 (A1 $_{hex}$) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	USINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 284: Input fieldbus variable USINT Extended 1 (A1 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert



12.4.5.31 Input Fieldbus Variable USINT Extended 2 (A2 hex)

Die Erweiterung der Klasse „Input Fieldbus Variable USINT" ermöglicht das Lesen von SPS-Eingangsvariablendaten. Der Instanzbereich der Klasse "Input Fieldbus Variable USINT Extended 2" deckt die SPS-Eingangsvariablendaten von 511 bis 512 ab.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Eingangsvariablen %IB3062...%IB3063.

Instanz 0 (Klassenattribute)

Tabelle 285: Input Fieldbus Variable USINT Extended 2 (A2 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	2 (0x0002)

Instanz 511...512 (511. bis 512. Eingangsvariable)

Tabelle 286: Input Fieldbus Variable USINT Extended 2 (A2 hex) – Instanz 511...512

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	USINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 287: Input fieldbus variable USINT Extended 2 (A2 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 hex	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.4.5.32 Output Fieldbus Variable USINT (A3 $_{hex}$)

Diese Klasse ermöglicht den Austausch von Daten einer bestimmten SPS-Ausgangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Ausgangsvariablen %QB2552...%QB2806.

Instanz 0 (Klassenattribute)

Tabelle 288: Output Fieldbus Variable USINT (A3 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 1...255 (1. bis 255. Ausgangsvariable)

Tabelle 289: Output Fieldbus Variable USINT (A3 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	USINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 290: Output Fieldbus Variable USINT (A3 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes



12.4.5.33 Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$)

Die Erweiterung der Klasse „Output Fieldbus Variable USINT“ ermöglicht den Austausch von SPS-Ausgangsvariablendaten. Der Instanzbereich der Klasse “Output Fieldbus Variable USINT Extended 1“ deckt die SPS-Ausgangsvariablendaten von 256 bis 510 ab.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Ausgangsvariablen %QB2807...%QB3061.

Instanz 0 (Klassenattribute)

Tabelle 291: Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 256...510 (256. bis 510. Ausgangsvariable)

Tabelle 292: Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	USINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 293: Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.34 Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$)

Die Erweiterung der Klasse „Output Fieldbus Variable USINT" ermöglicht den Austausch von SPS-Ausgangsvariablendaten. Der Instanzbereich der Klasse "Output Fieldbus Variable USINT Extended 2" deckt die SPS-Ausgangsvariablendaten von 511 bis 512 ab.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Ausgangsvariablen %QB3062...%QB3063.

Instanz 0 (Klassenattribute)

Tabelle 294: Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	2 (0x0002)

Instanz 511...512 (511. bis 512. Ausgangsvariable)

Tabelle 295: Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$) – Instanz 511...512

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	USINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 296: Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes



12.4.5.35 Input Fieldbus Variable UINT (A6 $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten einer bestimmten SPS-Eingangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Eingangsvariablen %IW1276...%IW1530.

Instanz 0 (Klassenattribute)

Tabelle 297: Input Fieldbus Variable UINT (A6 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 1...255 (1. bis 255. Eingangsvariable)

Tabelle 298: Input Fieldbus Variable UINT (A6 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	UINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 299: Input Fieldbus Variable UINT (A6 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert



12.4.5.36 Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$)

Die Erweiterung der Klasse „Input Fieldbus Variable UINT“ ermöglicht das Lesen von SPS-Eingangsvariablendaten. Der Instanzbereich der Klasse “Input Fieldbus Variable UINT Extended 1“ deckt die SPS-Eingangsvariablendaten von der SPS-Eingangsvariablen 256 ab.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys der SPS-Adresse für die Eingangsvariable %IW1531.

Instanz 0 (Klassenattribute)

Tabelle 300: Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	1 (0x0001)

Instanz 256 (256. Eingangsvariable)

Tabelle 301: Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$) – Instanz 256

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	UINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 302: Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert



12.4.5.37 Output Fieldbus Variable UINT (A8 $_{hex}$)

Diese Klasse ermöglicht den Austausch von Daten einer bestimmten SPS-Ausgangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Ausgangsvariablen %QW1276...%QW1530.

Instanz 0 (Klassenattribute)

Tabelle 303: Output Fieldbus Variable UINT (A8 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 1...255 (1. bis 255. Ausgangsvariable)

Tabelle 304: Output Fieldbus Variable UINT (A8 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	UINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 305: Output Fieldbus Variable UINT (A8 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.38 Output Fieldbus Variable UINT Extended 1 (A9 $_{hex}$)

Die Erweiterung der Klasse „Output Fieldbus Variable UINT“ ermöglicht den Austausch von SPS-Ausgangsvariablendaten. Der Instanzbereich der Klasse “Output Fieldbus Variable UINT Extended 1“ deckt die SPS-Ausgangsvariablendaten von der SPS-Ausgangsvariablen 256 ab.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Ausgangsvariable %QW1531.

Instanz 0 (Klassenattribute)

Tabelle 306: Output Fieldbus Variable UINT Extended 1 (A9 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	1 (0x0001)

Instanz 256 (256. Ausgangsvariable)

Tabelle 307: Output Fieldbus Variable UINT Extended 1 (A9 $_{hex}$) – Instanz 256

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	UINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 308: Output Fieldbus Variable UINT Extended 1 (A9 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes



12.4.5.39 Input Fieldbus Variable UDINT (AA $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten einer bestimmten SPS-Eingangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Eingangsvariablen %ID638 ... %ID765.

Instanz 0 (Klassenattribute)

Tabelle 309: Input Fieldbus Variable UDINT (AA $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	128 (0x0080)

Instanz 1...128 (1. bis 128. Eingangsvariable)

Tabelle 310: Input Fieldbus Variable UDINT (AA $_{hex}$) – Instanz 1...128

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	UDINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 311: Input Fieldbus Variable UDINT (AA $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.4.5.40 Input Fieldbus Variable UDINT Offset (AB $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten einer bestimmten SPS-Eingangsvariablen.
Mit einem Offset von 2 Byte auf die Adressen der Klasse “Input Fieldbus Variable UDINT (AA $_{hex}$)“ entspricht dieses bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Eingangsvariablen %ID638 ... %ID765.

Information



Informationen zur Verwendung des Offsets
Unter „Offset von 2 Byte“ ist Folgendes zu verstehen:
Wird Instanz 1 dieser Klasse ausgelesen, so erhalten Sie das High-Word der Adresse %ID638 und das Low-Word der Adresse %ID639, usw.
Wird Instanz 128 ausgelesen, so erhalten Sie nur das High-Word der Adresse %ID765.

Instanz 0 (Klassenattribute)

Tabelle 312: Input Fieldbus Variable UDINT Offset (AB $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	128 (0x080)

Instanz 1...128 (1. bis 128. Eingangsvariable)

Tabelle 313: Input Fieldbus Variable UDINT Offset (AB $_{hex}$) – Instanz 1...128

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	UDINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 314: Input Fieldbus Variable UDINT Offset (AB $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert



12.4.5.41 Output Fieldbus Variable UDINT (AC $_{hex}$)

Diese Klasse ermöglicht den Austausch von Daten einer bestimmten SPS-Ausgangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die Ausgangsvariablen %QD638 ... %QD765.

Instanz 0 (Klassenattribute)

Tabelle 315: Output Fieldbus Variable UDINT (AC $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	128 (0x0080)

Instanz 1...128 (1. bis 128. Ausgangsvariable)

Tabelle 316: Output Fieldbus Variable UDINT (AC $_{hex}$) – Instanz 1...128

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_Out_Var	UDINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 317: Output Fieldbus Variable UDINT (AC $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.4.5.42 Output Fieldbus Variable UDINT Offset (AD $_{hex}$)

Diese Klasse ermöglicht den Austausch von Daten einer bestimmten SPS-Ausgangsvariablen.
Mit einem Offset von 2 Byte auf die Adressen der Klasse "Output Fieldbus Variable UDINT (AC $_{hex}$)" entspricht dieses bei WAGO-I/O-*PRO* CAA bzw. CoDeSys den SPS-Adressen für die %QD638 ... %QD765.

Information



Informationen zur Verwendung des Offsets
Unter „Offset von 2 Byte" ist Folgendes zu verstehen:
Wird Instanz 1 dieser Klasse ausgelesen, so erhalten Sie das High-Word der Adresse %ID638 und das Low-Word der Adresse %ID639, usw.
Wird Instanz 128 ausgelesen, so erhalten Sie nur das High-Word der Adresse %ID765.

Instanz 0 (Klassenattribute)

Tabelle 318: Output Fieldbus Variable UDINT Offset (AD $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	128 (0x0080)

Instanz 1...128 (1. bis 128. Ausgangsvariable)

Tabelle 319: Output Fieldbus Variable UDINT Offset (AD $_{hex}$) – Instanz 1...128

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_Out_Var	UDINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 320: Output Fieldbus Variable UDINT Offset (AD $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes



13 Busklemmen

13.1 Übersicht

Für den Aufbau von Applikationen mit dem WAGO-I/O-SYSTEM 750 sind verschiedene Arten von Busklemmen verfügbar:

- Digitaleingangsklemmen
- Digitalausgangsklemmen
- Analogeingangsklemmen
- Analogausgangsklemmen
- Sonderklemmen
- Systemklemmen

Eine detaillierte Beschreibung zu jeder Busklemme und deren Varianten entnehmen Sie bitte den Handbüchern zu den Busklemmen.

Sie finden diese Beschreibungen auf der DVD-ROM „AUTOMATION Tools and Docs“ (Art. Nr.: 0888-0412) oder auf der Internetseite http://www.wago.com unter Dokumentation.

Information



Weitere Information zum WAGO-I/O-SYSTEM
Aktuelle Informationen zum modularen WAGO-I/O-SYSTEM finden Sie auf der Internetseite http://www.wago.com.

13.2 Aufbau der Prozessdaten für MODBUS/TCP

Der Aufbau der Prozessdaten ist bei einigen Busklemmen bzw. deren Varianten feldbusspezifisch.

Bei dem Feldbuscontroller mit MODBUS/TCP wird das Prozessabbild wortweise aufgebaut (mit word-alignment). Die interne Darstellung der Daten, die größer als ein Byte sind, erfolgt nach dem Intel-Format.

Im Folgenden wird für alle Busklemmen des WAGO-I/O-SYSTEM 750 und 753 die feldbusspezifische Darstellung im Prozessabbild des Feldbuscontrollers mit MODBUS/TCP beschrieben und der Aufbau der Prozesswerte gezeigt.

Für das Prozessabbild des Feldbuscontrollers ist der Aufbau der Prozesswerte identisch.

ACHTUNG



Geräteschäden durch falsche Adressierung!
Zur Vermeidung von Geräteschäden im Feldbereich, müssen Sie bei der Adressierung einer an beliebiger Position im Feldbusknoten befindlichen Busklemme, die Prozessdaten aller vorherigen byte- bzw. bitweise-orientierten Busklemmen berücksichtigen.



13.2.1 Digitaleingangsklemmen

Die Digitaleingangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Signalzustand des jeweiligen Kanals angibt. Diese Bits werden in das Eingangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie z.B. Drahtbruch und/oder Kurzschluss.

Sofern in dem Knoten auch Analogeingangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Eingangsdaten in dem Eingangsprozessabbild angehängt.

13.2.1.1 1-Kanal-Digitaleingangsklemmen mit Diagnose

750-435

Tabelle 321: 1-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnosebit S 1	Datenbit DI 1

13.2.1.2 2-Kanal-Digitaleingangsklemmen

750-400, -401, -405, -406, -410, -411, -412, -427, -438, (und alle Varianten),
753-400, -401, -405, -406, -410, -411, -412, -427

Tabelle 322: 2-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

13.2.1.3 2-Kanal-Digitaleingangsklemmen mit Diagnose

750-419, -421, -424, -425
753-421, -424, -425

Tabelle 323: 2-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

13.2.1.4 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

750-418
753-418

Die Digitaleingangsklemme liefert über die Prozesswerte im Eingangsprozessabbild hinaus 4 Bit Daten, die im Ausgangsprozessabbild dargestellt werden.

Tabelle 324: 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Ausgangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Quittierungs-bit Q 2 Kanal 2	Quittierungs-bit Q 1 Kanal 1	0	0

13.2.1.5 4-Kanal-Digitaleingangsklemmen

750-402, -403, -408, -409, -414, -415, -422, -423, -428, -432, -433, -1420, -1421, -1422
753-402, -403, -408, -409, -415, -422, -423, -428, -432, -433, -440

Tabelle 325: 4-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

13.2.1.6 8-Kanal-Digitaleingangsklemmen

750-430, -431, -436, -437, -1415, -1416, -1417
753-430, -431, -434

Tabelle 326: 8-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1



13.2.1.7 16-Kanal-Digitaleingangsklemmen

750-1400, -1402, -1405, -1406, -1407

Tabelle 327: 16-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 16 Kanal 16	Datenbit DI 15 Kanal 15	Datenbit DI 14 Kanal 14	Datenbit DI 13 Kanal 13	Datenbit DI 12 Kanal 12	Datenbit DI 11 Kanal 11	Datenbit DI 10 Kanal 10	Datenbit DI 9 Kanal 9	Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

13.2.2 Digitalausgangsklemmen

Die Digitalausgangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Status des jeweiligen Kanals angibt. Diese Bits werden in das Ausgangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie Drahtbruch und/oder Kurzschluss. Bei einigen Busklemmen müssen, bei gesetztem Diagnosebit, zusätzlich die Datenbits ausgewertet werden.

Sofern in dem Knoten auch Analogausgangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Ausgangsdaten in dem Ausgangsprozessabbild angehängt.

13.2.2.1 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

750-523

Die Digitalausgangsklemmen liefern über das eine Prozesswert-Bit im Ausgangsprozessabbild hinaus 1 Bit, das im Eingangsprozessabbild dargestellt wird. Dieses Statusbit zeigt den „Handbetrieb“ an.

Tabelle 328: 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	Statusbit “Hand-betrieb“

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	steuert DO 1 Kanal 1

13.2.2.2 2-Kanal-Digitalausgangsklemmen

750-501, -502, -509, -512, -513, -514, -517, -535, (und alle Varianten),
753-501, -502, -509, -512, -513, -514, -517

Tabelle 329: 2-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1



13.2.2.3 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-507 (-508), -522,
753-507

Die Digitalausgangsklemmen liefern über die 2-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 2 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 330: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

750-506,
753-506

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die durch einen 2-Bit-Fehlercode eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 331: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 75x-506

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 3 Kanal 2	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Diagnosebit S 0 Kanal 1

Diagnosebits S1/S0, S3/S2: = ‘00’ normaler Betrieb
Diagnosebits S1/S0, S3/S2: = ‘01’ keine Last angeschlossen/Kurzschluss gegen +24 V
Diagnosebits S1/S0, S3/S2: = ‘10’ Kurzschluss gegen GND/Überlast

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				nicht genutzt	nicht genutzt	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.4 4-Kanal-Digitalausgangsklemmen

750-504, -516, -519, -531,
753-504, -516, -531, -540

Tabelle 332: 4-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.5 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-532

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 333: 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Diagnosebit S = ‘0’ kein Fehler
Diagnosebit S = ‘1’ Drahtbruch, Kurzschluss oder Überlast

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.6 8-Kanal-Digitalausgangsklemmen

750-530, -536, -1515, -1516
753-530, -534

Tabelle 334: 8-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1



13.2.2.7 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-537

Die Digitalausgangsklemmen liefern über die 8-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 8 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 335: 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Diagnosebit S 8 Kanal 8	Diagnosebit S 7 Kanal 7	Diagnosebit S 6 Kanal 6	Diagnosebit S 5 Kanal 5	Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Diagnosebit S = ‘0’ kein Fehler
Diagnosebit S = ‘1’ Drahtbruch, Kurzschluss oder Überlast

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.8 16-Kanal-Digitalausgangsklemmen

750-1500, -1501, -1504, -1505

Tabelle 336: 16-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 16 Kanal 16	steuert DO 15 Kanal 15	steuert DO 14 Kanal 14	steuert DO 13 Kanal 13	steuert DO 12 Kanal 12	steuert DO 11 Kanal 11	steuert DO 10 Kanal 10	steuert DO 9 Kanal 9	steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.9 8-Kanal-Digitaleingangsklemmen/-Digitalausgangsklemmen

750-1502, -1506

Tabelle 337: 8-Kanal-Digitalein-/ -ausgangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1



13.2.3 Analogeingangsklemmen

Die Analogeingangsklemmen liefern je Kanal 16-Bit-Messwerte und 8 Steuer-/ Statusbits.

MODBUS/TCP verwendet die 8 Steuer-/ Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Eingangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/ -controller mit MODBUS/TCP deshalb nur die 16-Bit-Messwerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitaleingangsklemmen gesteckt sind, werden die analogen Eingangsdaten immer vor die digitalen Daten in das Eingangs-prozessabbild abgebildet.

Information



Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Homepage unter: http://www.wago.com.

13.2.3.1 1-Kanal-Analogeingangsklemmen

750-491, (und alle Varianten)

Tabelle 338: 1-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Messwert U_D
1	D3	D2	Messwert U_{ref}

13.2.3.2 2-Kanal-Analogeingangsklemmen

750-452, -454, -456, -461, -462, -465, -466, -467, -469, -472, -474, -475, 476, -477, -478, -479, -480, -481, -483, -485, -492, (und alle Varianten),
753-452, -454, -456, -461, -465, -466, -467, -469, -472, -474, -475, 476, -477, 478, -479, -483, -492, (und alle Varianten)

Tabelle 339: 2-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Messwert Kanal 1
1	D3	D2	Messwert Kanal 2

13.2.3.3 4-Kanal-Analogeingangsklemmen

750-453, -455, -457, -459, -460, -468, (und alle Varianten),
753-453, -455, -457, -459

Tabelle 340: 4-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Messwert Kanal 1
1	D3	D2	Messwert Kanal 2
2	D5	D4	Messwert Kanal 3
3	D7	D6	Messwert Kanal 4



13.2.4 Analogausgangsklemmen

Die Analogausgangsklemmen liefern je Kanal 16-Bit-Ausgabewerte und 8 Steuer-/Statusbits.
MODBUS/TCP verwendet die 8 Steuer-/Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Ausgangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/-controller mit MODBUS/TCP deshalb nur die 16-Bit-Ausgabewerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitalausgangsklemmen gesteckt sind, werden die analogen Ausgangsdaten immer vor die digitalen Daten in das Ausgangs-prozessabbild abgebildet.

Information



Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Homepage unter: http://www.wago.com.

13.2.4.1 2-Kanal-Analogausgangsklemmen

750-550, -552, -554, -556, -560, -562, 563, -585, (und alle Varianten),
753-550, -552, -554, -556

Tabelle 341: 2-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Ausgabewert Kanal 1
1	D3	D2	Ausgabewert Kanal 2

13.2.4.2 4-Kanal-Analogausgangsklemmen

750-553, -555, -557, -559,
753-553, -555, -557, -559

Tabelle 342: 4-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Ausgabewert Kanal 1
1	D3	D2	Ausgabewert Kanal 2
2	D5	D4	Ausgabewert Kanal 3
3	D7	D6	Ausgabewert Kanal 4

13.2.5 Sonderklemmen

Bei einzelnen Klemmen wird neben den Datenbytes auch das Control-/Statusbyte eingeblendet. Dieses dient dem bidirektionalen Datenaustausch der Busklemme mit der übergeordneten Steuerung.

Das Controlbyte wird von der Steuerung an die Klemme und das Statusbyte von der Klemme an die Steuerung übertragen. Somit ist beispielsweise das Setzen eines Zählers mit dem Steuerbyte oder die Anzeige von Bereichsunter- oder -überschreitung durch das Statusbyte möglich.

Das Control-/Statusbyte liegt im Prozessabbild stets im Low-Byte.

Information



Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der Internetseite http://www.wago.com.

13.2.5.1 Zählerklemmen

750-404, (und alle Varianten außer /000-005),
753-404, (und Variante /000-003)

Die Zählerklemmen belegen insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie ein zusätzliches Steuer-/ Statusbyte. Die Busklemmen liefern dann 32-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 343: Zählerklemmen 750-404, (und alle Varianten außer /000-005), 753-404, (und Variante /000-003)

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	S	Statusbyte
1	D1	D0	Zählerwert
2	D3	D2	

Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C	Steuerbyte
1	D1	D0	Zählersetzwert
2	D3	D2	



750-404/000-005

Die Zählerklemmen belegen insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich der Prozessabbilder, 4 Datenbytes sowie ein zusätzliches Steuer-/Statusbyte. Diese Busklemmen liefern pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 344: Zählerklemmen 750-404/000-005

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	S	Statusbyte
1	D1	D0	Zählerwert Zähler 1
2	D3	D2	Zählerwert Zähler 2

Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C	Steuerbyte
1	D1	D0	Zählersetzwert Zähler 1
2	D3	D2	Zählersetzwert Zähler 2

750-638,
753-638

Diese Zählerklemmen belegen insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Die Busklemmen liefern dann pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 345: Zählerklemmen 750-638, 753-638

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	S0	Statusbyte von Zähler 1
1	D1	D0	Zählerwert von Zähler 1
2	-	S1	Statusbyte von Zähler 2
3	D3	D2	Zählerwert von Zähler 2

Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C0	Steuerbyte von Zähler 1
1	D1	D0	Zählersetzwert von Zähler 1
2	-	C1	Steuerbyte von Zähler 2
3	D3	D2	Zählersetzwert von Zähler 2

13.2.5.2 Pulsweitenklemmen

750-511, (und alle Varianten /xxx-xxx)

Diese Pulsweitenklemmen belegen insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 346: Pulsweitenklemmen 750-511, /xxx-xxx

Ein- und Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C0/S0	Steuer-/Statusbyte von Kanal 1
1	D1	D0	Datenwert von Kanal 1
2	-	C1/S1	Steuer-/Statusbyte von Kanal 2
3	D3	D2	Datenwert von Kanal 2

13.2.5.3 Serielle Schnittstellen mit alternativem Datenformat

750-650, (und die Varianten /000-002, -004, -006, -009, -010, -011, -012, -013),
750-651, (und die Varianten /000-001, -002, -003),
750-653, (und die Varianten /000-002, -007),

753-650, -653

Hinweis



Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrierten Betriebsart!
Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Der Aufbau des Prozessabbilds dieser Busklemme hängt dann davon ab, welche Betriebsart eingestellt ist.

Die seriellen Schnittstellenklemmen, die auf das alternative Datenformat eingestellt sind, belegen insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 347: Serielle Schnittstellen mit alternativem Datenformat

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	C/S	Datenbyte	Steuer-/Statusbyte
1	D2	D1	Datenbytes	



13.2.5.4 Serielle Schnittstellen mit Standard-Datenformat

750-650/000-001, -014, -015, -016
750-653/000-001, -006

Die seriellen Schnittstellenklemmen, die auf das Standard-Datenformat eingestellt sind, belegen insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches
Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 348: Serielle Schnittstellen mit Standard-Datenformat

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	D0	C/S	Datenbyte	Steuer-/ Statusbyte
1	D2	D1	Datenbytes	
2	D4	D3		

13.2.5.5 Datenaustauschklemmen

750-654, (und die Variante /000-001)

Die Datenaustauschklemmen belegen jeweils insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 349: Datenaustauschklemmen

Ein- und Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Datenbytes
1	D3	D2	

13.2.5.6 SSI-Geber-Interface-Busklemmen

750-630, (und alle Varianten)

Hinweis



Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrierten Betriebsart!
Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Der Aufbau des Prozessabbilds dieser Busklemme hängt dann davon ab, welche Betriebsart eingestellt ist.

Die SSI-Geber Interface Busklemmen mit Status belegen insgesamt 4 Datenbytes im Eingangsbereich des Prozessabbilds. Dabei werden mit word-alignment insgesamt 2 Worte im Prozessabbild belegt.

Tabelle 350: SSI-Geber Interface Busklemmen mit alternativem Datenformat

Eingangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Datenbytes
1	D3	D2	

13.2.5.7 Weg- und Winkelmessung

750-631/000-004, -010, -011

Die Busklemme 750-631 belegt 5 Bytes im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 351: Weg- und Winkelmessung 750-631/000-004, --010, -011

Eingangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	-	S	nicht genutzt	Statusbyte
1	D1	D0	Zählerwort	
2	-	-	nicht genutzt	
3	D4	D3	Latchwort	

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C	Steuerbyte von Zähler 1
1	D1	D0	Zählersetzwert von Zähler 1
2	-	-	nicht genutzt
3	-	-	nicht genutzt

750-634

Die Busklemme 750-634 belegt 5 Bytes (in der Betriebsart Periodendauermessung mit 6 Bytes) im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.



Tabelle 352: Incremental Encoder Interface 750-634

Eingangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	-	S	nicht genutzt	Statusbyte
1	D1	D0	Zählerwort	
2	-	(D2) *)	nicht genutzt	(Periodendauer)
3	D4	D3	Latchwort	

*) Ist durch das Steuerbyte die Betriebsart Periodendauermessung eingestellt, wird in D2 zusammen mit D3/D4 die Periodendauer als 24-Bit-Wert ausgegeben.

Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	-	C	nicht genutzt	Steuerbyte
1	D1	D0	Zählersetzwort	
2	-	-	nicht genutzt	
3	-	-		

750-637

Die Inkremental Encoder Interface Busklemme belegt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes und zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 353: Inkremental Encoder Interface 750-637

Ein- und Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	-	C0/S0	Steuer-/Statusbyte von Kanal 1
1	D1	D0	Datenwerte von Kanal 1
2	-	C1/S1	Steuer-/Statusbyte von Kanal 2
3	D3	D2	Datenwerte von Kanal 2

750-635,
753-635

Die Digitale Impuls Schnittstelle belegt insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 354: Digitale Impuls Schnittstelle 750-635

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	C0/S0	Datenbyte	Steuer-/Statusbyte
1	D2	D1	Datenbytes	

13.2.5.8 DC-Drive Controller

750-636

Der DC-Drive-Controller 750-636 stellt dem Koppler über 1 logischen Kanal 6 Byte Ein- und Ausgangsprozessabbild zur Verfügung. Die zu sendenden und zu empfangenden Positionsdaten werden in 4 Ausgangsbytes (D0 ... D3) und 4 Eingangsbytes (D0 ... D3) abgelegt. 2 Steuerbytes (C0, C1) und 2 Statusbytes (S0, S1) dienen zur Steuerung der Busklemme und des Antriebs. Alternativ zu den Positionsdaten im Eingangsprozessabbild (D0 ... D3) können erweiterte Statusinformationen (S2 ... S5) eingeblendet werden. Die 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen den Prozessdaten und den erweiterten Statusbytes im Eingangsprozessabbild erfolgt über Bit 3 (ExtendedInfo_ON) im Controlbyte C1 (C1.3). Mit Bit 3 des Statusbytes S1 (S1.3) wird die Umschaltung quittiert.

Tabelle 355: Antriebssteuerung 750-636

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	S1	S0	Status S1	Statusbyte S0
1	D1*) / S3**)	D0*) / S2**)	Istposition*) / Erweitertes Statusbyte S3**)	Istposition (LSB)*) / Erweitertes Statusbyte S2**)
2	D3*) / S5**)	D2*) / S4**)	Istposition (MSB)*) / Erweitertes Statusbyte S3**)	Istposition*) / Erweitertes Statusbyte S4**)

*) ExtendedInfo_ON = ‘0’.
**) ExtendedInfo_ON = ‘1’.



Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	C1	C0	Steuerbyte C1	Steuerbyte C0
1	D1	D0	Sollposition	Sollposition (LSB)
2	D3	D2	Sollposition (MSB)	Sollposition

13.2.5.9 Steppercontroller

750-670

Der Steppercontroller RS 422 / 24 V / 20 mA 750-670 stellt dem Feldbuskoppler über 1 logischen Kanal 12 Byte Ein- und Ausgangsprozessabbild zur Verfügung. Die zu sendenden und zu empfangenden Daten werden in Abhängigkeit von der Betriebsart in bis zu 7 Ausgangsbytes (D0 ... D6) und 7 Eingangsbytes (D0 ... D6) abgelegt. Das Ausgangsbyte D0 und das Eingangsbyte D0 sind reserviert und ohne Funktion. Ein Klemmenbus-Steuer- und Statusbyte (C0, S0) sowie 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen beiden Prozessabbildern erfolgt über das Bit 5 im Controlbyte C0 (C0.5). Mit dem Bit 5 des Statusbytes S0 (S0.5) wird das Einschalten der Mailbox quittiert.

Tabelle 356: Steppercontroller RS 422 / 24 V / 20 mA 750-670

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	Reserviert	S0	Reserviert	Statusbyte S0
1	D1	D0	Prozessdaten*) / Mailbox**)	
2	D3	D2		
3	D5	D4		
4	S3	D6	Statusbyte S3	Prozessdaten*) / Reserviert**)
5	S1	S2	Statusbyte S1	Statusbyte S2

*) Zyklisches Prozessabbild (Mailbox ausgeschaltet).
**) Mailboxprozessabbild (Mailbox eingeschaltet)

Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	Reserviert	C0	Reserviert	Controlbyte C0
1	D1	D0	Prozessdaten*) / Mailbox**)	
2	D3	D2		
3	D5	D4		
4	C3	D6	Controlbyte C3	Prozessdaten*) / Reserviert**)
5	C1	C2	Controlbyte C1	Controlbyte C2

*) Zyklisches Prozessabbild (Mailbox ausgeschaltet).
**) Mailboxprozessabbild (Mailbox eingeschaltet)

13.2.5.10 RTC-Modul

750-640

Das RTC-Modul belegt insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes, ein zusätzliches Steuer-/ Statusbyte und jeweils ein Befehlsbyte (ID). Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 357: RTC-Modul 750-640

Ein- und Ausgangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	ID	C/S	Befehlsbyte	Steuer-/ Statusbyte
1	D1	D0	Datenbytes	
2	D3	D2		

13.2.5.11 DALI/DSI-Masterklemme

750-641

Die DALI/DSI-Masterklemme belegt insgesamt 6 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches Steuer- /Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 358: DALI/DSI-Masterklemme 750-641

Eingangsprozessabbild				
Offset	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
0	D0	S	DALI-Antwort	Statusbyte
1	D2	D1	Message 3	DALI-Adresse
2	D4	D3	Message 1	Message 2



Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	C	DALI-Befehl, DSI-Dimmwert	Steuerbyte
1	D2	D1	Parameter 2	DALI-Adresse
2	D4	D3	Command-Extension	Parameter 1

13.2.5.12 Funkreceiver EnOcean

750-642

Die EnOcean Funkreceiverklemme belegt insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Die 3 Bytes Ausgangsdaten werden jedoch nicht genutzt. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 359: Funkreceiver EnOcean 750-642

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	S	Datenbyte	Statusbyte
1	D2	D1	Datenbytes	

Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	C	nicht genutzt	Steuerbyte
1	-	-	nicht genutzt	

13.2.5.13 MP-Bus-Masterklemme

750-643

Die MP-Bus-Masterklemme belegt insgesamt 8 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbildes, 6 Datenbytes und zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 360: MP-Bus-Masterklemme 750-643

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	C1/S1	C0/S0	erweitertes Steuer-/Statusbyte	Steuer-/Statusbyte
1	D1	D0	Datenbytes	
2	D3	D2		
3	D5	D4		

13.2.5.14 *Bluetooth*® RF-Transceiver

750-644

Die Größe des Prozessabbildes der *Bluetooth*®-Busklemme ist in den festgelegten Größen 12, 24 oder 48 Byte einstellbar.
Es besteht aus einem Steuerbyte (Eingang) bzw. Statusbyte (Ausgang), einem Leerbyte, einer 6, 12 oder 18 Byte großen, überlagerbaren Mailbox (Modus 2) und den *Bluetooth*®-Prozessdaten in einem Umfang von 4 bis 46 Byte.

Die *Bluetooth*®-Busklemme belegt also jeweils 12 bis maximal 48 Bytes im Prozessabbild, wobei die Größen des Eingangs- und Ausgangsprozessabbildes stets übereinstimmen.

Das erste Byte enthält das Steuer-/Statusbyte, das zweite ein Leerbyte.
Daran schließen sich bei ausgeblendeter Mailbox unmittelbar Prozessdaten an.
Bei eingeblendeter Mailbox werden je nach deren Größe die ersten 6, 12 oder 18 Byte Prozessdaten von Mailbox-Daten überlagert. Die Bytes im Bereich hinter der optional einblendbaren Mailbox enthalten grundsätzlich Prozessdaten. Den internen Aufbau der *Bluetooth*®-Prozessdaten entnehmen Sie der Dokumentation des *Bluetooth*® RF-Transceivers 750-644.

Die Einstellung der Mailbox- und Prozessabbildgrößen erfolgt mit dem Inbetriebnahmetool WAGO-I/O-*CHECK*.

Tabelle 361: Bluetooth® RF-Transceiver 750-644

<table>
<tr><th colspan="5">Ein- und Ausgangsprozessabbild</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">Bezeichnung der Bytes</th><th rowspan="2" colspan="2">Bemerkung</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>-</td><td>C0/S0</td><td>nicht genutzt</td><td>Steuer-/ Statusbyte</td></tr>
<tr><td>1</td><td>D1</td><td>D0</td><td rowspan="5" colspan="2">Mailbox (0, 3, 6 oder 9 Worte) sowie Prozessdaten (2-23 Worte)</td></tr>
<tr><td>2</td><td>D3</td><td>D2</td></tr>
<tr><td>3</td><td>D5</td><td>D4</td></tr>
<tr><td>...</td><td>...</td><td>...</td></tr>
<tr><td>max. 23</td><td>D45</td><td>D44</td></tr>
</table>



13.2.5.15 Schwingstärke/Wälzlagerüberwachung VIB I/O

750-645

Die Schwingstärke/Wälzlagerüberwachung VIB I/O belegt insgesamt 12 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 8 Datenbytes und vier zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 8 Worte im Prozessabbild belegt.

Tabelle 362: Schwingstärke/Wälzlagerüberwachung VIB I/O 750-645

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	C0/S0	nicht genutzt	Steuer-/Statusbyte (log. Kanal 1, Sensoreingang 1)
1	D1	D0	Datenbytes (log. Kanal 1, Sensoreingang 1)	
2	-	C1/S1	nicht genutzt	Steuer-/Statusbyte (log. Kanal 2, Sensoreingang 2)
3	D3	D2	Datenbytes (log. Kanal 2, Sensoreingang 2)	
4	-	C2/S2	nicht genutzt	Steuer-/Statusbyte (log. Kanal 3, Sensoreingang 3)
5	D5	D4	Datenbytes (log. Kanal 3, Sensoreingang 3)	
6	-	C3/S3	nicht genutzt	Steuer-/Statusbyte (log. Kanal 4, Sensoreingang 4)
7	D7	D6	Datenbytes (log. Kanal 4, Sensoreingang 4)	

13.2.5.16 KNX/EIB/TP1-Klemme

753-646

Die KNX/TP1-Klemme erscheint im Router- sowie im Gerätemodus mit insgesamt 24 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbildes, 20 Datenbytes und 1 Steuer-/Statusbyte. Die zusätzlichen Bytes S1 bzw. C1 werden als Datenbytes transferiert, aber als erweiterte Status- und Steuerbytes verwendet. Der Opcode dient als Schreib- und Lesekommando für Daten oder als Auslöser bestimmter Funktionen der KNX/EIB/TP1-Klemme. Mit word-alignment werden jeweils 12 Worte im Prozessabbild belegt. Im Routermodus ist kein Zugriff auf das Prozessabbild möglich. Telegramme werden nur getunnelt übertragen.

Im Gerätemodus erfolgt der Zugriff auf KNX-Daten über spezielle Funktionsbausteine der IEC-Applikation. Eine Konfiguration mittels der allgemeinen Engineering-Tool-Software (ETS) für KNX ist notwendig.



Tabelle 363: KNX/EIB/TP1-Klemme 753-646

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	S0	nicht genutzt	Statusbyte
1	S1	OP	Erweitertes Statusbyte	Opcode
2	D1	D0	Datenbyte 1	Datenbyte 0
3	D3	D2	Datenbyte 3	Datenbyte 2
4	D5	D4	Datenbyte 5	Datenbyte 4
5	D7	D6	Datenbyte 7	Datenbyte 6
6	D9	D8	Datenbyte 9	Datenbyte 8
7	D11	D10	Datenbyte 11	Datenbyte 10
8	D13	D12	Datenbyte 13	Datenbyte 12
9	D15	D14	Datenbyte 15	Datenbyte 14
10	D17	D16	Datenbyte 17	Datenbyte 16
11	D19	D18	Datenbyte 19	Datenbyte 18

Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	C0	nicht genutzt	Steuerbyte
1	C1	OP	Erweitertes Steuerbyte	Opcode
2	D1	D0	Datenbyte 1	Datenbyte 0
3	D3	D2	Datenbyte 3	Datenbyte 2
4	D5	D4	Datenbyte 5	Datenbyte 4
5	D7	D6	Datenbyte 7	Datenbyte 6
6	D9	D8	Datenbyte 9	Datenbyte 8
7	D11	D10	Datenbyte 11	Datenbyte 10
8	D13	D12	Datenbyte 13	Datenbyte 12
9	D15	D14	Datenbyte 15	Datenbyte 14
10	D17	D16	Datenbyte 17	Datenbyte 16
11	D19	D18	Datenbyte 19	Datenbyte 18

13.2.5.17 AS-interface-Masterklemme

750-655

Das Prozessabbild der AS-interface-Masterklemme ist in seiner Länge einstellbar in den festgelegten Größen von 12, 20, 24, 32, 40 oder 48 Byte.
Es besteht aus einem Control- bzw. Statusbyte, einer 0, 6, 10, 12 oder 18 Byte großen Mailbox und den AS-interface Prozessdaten in einem Umfang von 0 bis 32 Byte.

Mit word-alignment belegt die AS-interface-Masterklemme also jeweils 6 bis maximal 24 Worte im Prozessabbild.

Das erste Ein- bzw. Ausgangswort enthält das Status- bzw. Controlbyte sowie ein Leerbyte.
Daran schließen sich für die fest eingeblendete Mailbox (Modus 1) die Worte mit Mailboxdaten an.



Wenn die Mailbox überlagerbar eingestellt ist (Modus 2), enthalten diese Worte Mailbox- oder Prozessdaten.
Die weiteren Worte enthalten die restlichen Prozessdaten.

Die Einstellung der Mailbox- und Prozessabbildgrößen erfolgt mit dem Inbetriebnahmetool WAGO-I/O-*CHECK*.

Tabelle 364: AS-interface-Masterklemme 750-655

<table>
<tr><th colspan="5">Ein- und Ausgangsprozessabbild</th></tr>
<tr><th rowspan="2">Offset</th><th colspan="2">Bezeichnung der Bytes</th><th colspan="2" rowspan="2">Bemerkung</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>0</td><td>-</td><td>C0/S0</td><td>nicht genutzt</td><td>Steuer-/Statusbyte</td></tr>
<tr><td>1</td><td>D1</td><td>D0</td><td colspan="2" rowspan="5">Mailbox (0, 3, 5, 6 oder 9 Worte) sowie Prozessdaten (0-16 Worte)</td></tr>
<tr><td>2</td><td>D3</td><td>D2</td></tr>
<tr><td>3</td><td>D5</td><td>D4</td></tr>
<tr><td>...</td><td>...</td><td>...</td></tr>
<tr><td>max. 23</td><td>D45</td><td>D44</td></tr>
</table>

13.2.6 Systemklemmen

13.2.6.1 Systemklemmen mit Diagnose

750-610, -611

Die Potentialeinspeiseklemmen 750-610 und -611 mit Diagnose liefern zur Überwachung der Versorgung 2 Bits Diagnosedaten.

Tabelle 365: Systemklemmen mit Diagnose 750-610, -611

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnosebit S 2 Sicherung	Diagnosebit S 1 Spannung

13.2.6.2 Binäre Platzhalterklemmen

750-622

Die binären Platzhalterklemmen 750-622 verhalten sich wahlweise wie 2-Kanal-Digitaleingangsklemmen oder -ausgangsklemmen und belegen je nach angewählter Einstellung pro Kanal 1, 2, 3 oder 4 Bits.
Dabei werden dann entsprechend 2, 4, 6 oder 8 Bits entweder im Prozesseingangs- oder ausgangsabbild belegt.

Tabelle 366: Binäre Platzhalterklemmen 750-622 (mit dem Verhalten einer 2 DI)

Ein- oder Ausgangsgangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
(Datenbit DI 8)	(Datenbit DI 7)	(Datenbit DI 6)	(Datenbit DI 5)	(Datenbit DI 4)	(Datenbit DI 3)	Datenbit DI 2	Datenbit DI 1



13.3 Aufbau der Prozessdaten für EtherNet/IP

Der Aufbau der Prozessdaten ist bei einigen Busklemmen bzw. deren Varianten feldbusspezifisch.

Bei dem Feldbuscontroller mit EtherNet/IP wird das Prozessabbild wortweise aufgebaut (mit word-alignment). Die interne Darstellung der Daten, die größer als ein Byte sind, erfolgt nach dem Intel-Format.

Im Folgenden wird für alle Busklemmen des WAGO-I/O-SYSTEM 750 und 753 die feldbusspezifische Darstellung im Prozessabbild des Feldbuscontrollers mit EtherNet/IP beschrieben und der Aufbau der Prozesswerte gezeigt.

Für das PFC-Prozessabbild des Controllers ist der Aufbau der Prozesswerte identisch.

ACHTUNG



Geräteschäden durch falsche Adressierung!
Zur Vermeidung von Geräteschäden im Feldbereich, müssen Sie bei der Adressierung einer an beliebiger Position im Feldbusknoten befindlichen Busklemme, die Prozessdaten aller vorherigen byte- bzw. bitweise-orientierten Busklemmen berücksichtigen.

13.3.1 Digitaleingangsklemmen

Die Digitaleingangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Signalzustand des jeweiligen Kanals angibt. Diese Bits werden in das Eingangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie z.B. Drahtbruch und/oder Kurzschluss.

Sofern in dem Knoten auch Analogeingangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Eingangsdaten in dem Eingangsprozessabbild angehängt.
Dabei wird für jeweils 8 Bit ein Subindex belegt.

Jeder Eingangskanal belegt in dem Discrete Input Point Object (Class 0x65) eine Instanz.

13.3.1.1 1-Kanal-Digitaleingangsklemmen mit Diagnose

750-435

Tabelle 367: 1-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnosebit S 1	Datenbit DI 1

Die Eingangsklemmen belegen in Klasse (0x65) 2 Instanzen.

13.3.1.2 2-Kanal-Digitaleingangsklemmen

750-400, -401, -405, -406, -410, -411, -412, -427, -438, (und alle Varianten),
753-400, -401, -405, -406, -410, -411, -412, -427

Tabelle 368: 2-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 2 Instanzen.



13.3.1.3 2-Kanal-Digitaleingangsklemmen mit Diagnose

750-419, -421, -424, -425
753-421, -424, -425

Tabelle 369: 2-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 4 Instanzen.

13.3.1.4 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

750-418
753-418

Die Digitaleingangsklemme 750-418, 753-418 liefert über die Prozesswerte im Eingangsprozessabbild hinaus 4 Bit Daten, die im Ausgangsprozessabbild dargestellt werden.

Tabelle 370: 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 4 Instanzen.

Ausgangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Quittierungsb it Q 2 Kanal 2	Quittierung sbit Q 1 Kanal 1	0	0

Die Eingangsklemmen belegen in Klasse (0x66) 4 Instanzen.

13.3.1.5 4-Kanal-Digitaleingangsklemmen

750-402, -403, -408, -409, -414, -415, -422, -423, -428, -432, -433, -1420, -1421, -1422
753-402, -403, -408, -409, -415, -422, -423, -428, -432, -433, -440

Tabelle 371: 4-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 4 Instanzen.

13.3.1.6 8-Kanal-Digitaleingangsklemmen

750-430, -431, -436, -437, -1415, -1416, -1417
753-430, -431, -434

Tabelle 372: 8-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 8 Instanzen.

13.3.1.7 16-Kanal-Digitaleingangsklemmen

750-1400, -1402, -1405, -1406, -1407

Tabelle 373: 16-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Daten bit DI 16 Kanal 16	Daten bit DI 15 Kanal 15	Daten bit DI 14 Kanal 14	Daten bit DI 13 Kanal 13	Daten bit DI 12 Kanal 12	Daten bit DI 11 Kanal 11	Daten bit DI 10 Kanal 10	Daten bit DI 9 Kanal 9	Daten bit DI 8 Kanal 8	Daten bit DI 7 Kanal 7	Daten bit DI 6 Kanal 6	Daten bit DI 5 Kanal 5	Daten bit DI 4 Kanal 4	Daten bit DI 3 Kanal 3	Daten bit DI 2 Kanal 2	Daten bit DI 1 Kanal 1

13.3.2 Digitalausgangsklemmen

Die Digitalausgangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Status des jeweiligen Kanals angibt. Diese Bits werden in das Ausgangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie Drahtbruch und/oder Kurzschluss. Bei einigen Busklemmen müssen, bei gesetztem Diagnosebit, zusätzlich die Datenbits ausgewertet werden.

Sofern in dem Knoten auch Analogausgangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Ausgangsdaten in dem Ausgangsprozessabbild angehängt.
Dabei wird für jeweils 8 Bit ein Subindex belegt.

Jeder Ausgangskanal belegt in dem Discrete Output Point Object (Class 0x66) eine Instanz.

13.3.2.1 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

750-523

Die Digitalausgangsklemmen liefern über das Prozesswerte Bit im Ausgangsprozessabbild hinaus 1 Bit, das im Eingangsprozessabbild dargestellt wird. Dieses Statusbit zeigt den "Handbetrieb" an.

Tabelle 374: 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	Statusbit “Hand-betrieb“

Die Ausgangsklemmen belegen in Klasse (0x65) 2 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 2 Instanzen.

13.3.2.2 2-Kanal-Digitalausgangsklemmen

750-501, -502, -509, -512, -513, -514, -517, -535, (und alle Varianten),
753-501, -502, -509, -512, -513, -514, -517

Tabelle 375: 2-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 2 Instanzen.

13.3.2.3 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-507 (-508), -522,
753-507

Die Digitalausgangsklemmen liefern über die 2-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 2 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 376: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x65) 2 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 2 Instanzen.

750-506,
753-506

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die durch einen 2-Bit Fehlercode eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.



Tabelle 377: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 75x-506

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 3 Kanal 2	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Diagnosebit S 0 Kanal 1
Diagnosebits S1/S0, S3/S2: = ‘00’ normaler Betrieb Diagnosebits S1/S0, S3/S2: = ‘01’ keine Last angeschlossen/Kurzschluss gegen +24 V Diagnosebits S1/S0, S3/S2: = ‘10’ Kurzschluss gegen GND/Überlast							

Die Ausgangsklemmen belegen in Klasse (0x65) 4 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				nicht genutzt	nicht genutzt	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 4 Instanzen.

13.3.2.4 4-Kanal-Digitalausgangsklemmen

750-504, -516, -519, -531,
753-504, -516, -531, -540

Tabelle 378: 4-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 4 Instanzen.

13.3.2.5 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-532

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 379: 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1
Diagnosebit S = ‘0’	kein Fehler						
Diagnosebit S = ‘1’	Drahtbruch, Kurzschluss oder Überlast						

Die Ausgangsklemmen belegen in Klasse (0x65) 4 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 4 Instanzen.

13.3.2.6 8-Kanal-Digitalausgangsklemmen

750-530, -536, -1515, -1516
753-530, -534

Tabelle 380: 8-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 8 Instanzen.



13.3.2.7 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-537

Die Digitalausgangsklemmen liefern über die 8-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 8 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 381: 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Diagnosebit S 8 Kanal 8	Diagnosebit S 7 Kanal 7	Diagnosebit S 6 Kanal 6	Diagnosebit S 5 Kanal 5	Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1
Diagnosebit S = ‘0’		kein Fehler					
Diagnosebit S = ‘1’		Drahtbruch, Kurzschluss oder Überlast					

Die Ausgangsklemmen belegen in Klasse (0x65) 8 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 8 Instanzen.

13.3.2.8 16-Kanal-Digitalausgangsklemmen

750-1500, -1501, -1504, -1505

Tabelle 382: 16-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 16 Kanal 16	steuert DO 15 Kanal 15	steuert DO 14 Kanal 14	steuert DO 13 Kanal 13	steuert DO 12 Kanal 12	steuert DO 11 Kanal 11	steuert DO 10 Kanal 10	steuert DO 9 Kanal 9	steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 16 Instanzen.

13.3.2.9 8-Kanal-Digitalein- / -ausgangsklemmen

750-1502, -1506

Tabelle 383: 8-Kanal-Digitalein-/ -ausgangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Ein-/Ausgangsklemmen belegen in Klasse (0x65) 8 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ein-/Ausgangsklemmen belegen in Klasse (0x66) 8 Instanzen.



13.3.3 Analogeingangsklemmen

Die Analogeingangsklemmen liefern je Kanal 16-Bit-Messwerte und 8 Steuer-/ Statusbits.

EtherNet/IP verwendet die 8 Steuer-/ Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Eingangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/ -controller mit EtherNet/IP deshalb nur die 16-Bit-Messwerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitaleingangsklemmen gesteckt sind, werden die analogen Eingangsdaten immer vor die digitalen Daten in das Eingangs-prozessabbild abgebildet.

Jeder Eingangskanal belegt in dem Analog Input Point Object (Class 0x67) eine Instanz.

Information



Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Internetseite unter: http://www.wago.com.

13.3.3.1 1-Kanal-Analogeingangsklemmen

750-491, (und alle Varianten)

Tabelle 384: 1-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Instanz	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
n	D1	D0	Messwert U_D
n+1	D3	D2	Messwert U_{ref}

Diese Eingangsklemmen stellen sich mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen.

13.3.3.2 2-Kanal-Analogeingangsklemmen

750-452, -454, -456, -461, -462, -465, -466, -467, -469, -472, -474, -475, 476, -477, -478, -479, -480, -481, -483, -485, -492, (und alle Varianten),
753-452, -454, -456, -461, -465, -466, -467, -469, -472, -474, -475, 476, -477, 478, -479, -483, -492, (und alle Varianten)

Tabelle 385: 2-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Messwert Kanal 1
n+1	D3	D2	Messwert Kanal 2

Diese Eingangsklemmen stellen sich mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen.

13.3.3.3 4-Kanal-Analogeingangsklemmen

750-453, -455, -457, -459, -460, -468, (und alle Varianten),
753-453, -455, -457, -459

Tabelle 386: 4-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Messwert Kanal 1
n+1	D3	D2	Messwert Kanal 2
n+2	D5	D4	Messwert Kanal 3
n+3	D7	D6	Messwert Kanal 4

Diese Eingangsklemmen stellen sich mit 4x2 Byte dar und belegen in Klasse (0x67) 4 Instanzen.



13.3.4 Analogausgangsklemmen

Die Analogausgangsklemmen liefern je Kanal 16-Bit-Ausgabewerte und 8 Steuer-/ Statusbits.
EtherNet/IP verwendet die 8 Steuer-/ Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Ausgangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/-controller mit EtherNet/IP deshalb nur die 16-Bit-Ausgabewerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitalausgangsklemmen gesteckt sind, werden die analogen Ausgangsdaten immer vor die digitalen Daten in das Ausgangs-prozessabbild abgebildet.

Jeder Ausgangskanal belegt in dem Analog Output Point Object (Class 0x68) eine Instanz.

Information



Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Internetseite unter: http://www.wago.com.

13.3.4.1 2-Kanal-Analogausgangsklemmen

750-550, -552, -554, -556, -560, -562, 563, -585, (und alle Varianten),
753-550, -552, -554, -556

Tabelle 387: 2-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Instanz	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
n	D1	D0	Ausgabewert Kanal 1
n+1	D3	D2	Ausgabewert Kanal 2

Diese Ausgangsklemmen stellen sich mit 2x2 Byte dar und belegen in Klasse (0x68) 2 Instanzen.

13.3.4.2 4-Kanal-Analogausgangsklemmen

750-553, -555, -557, -559,
753-553, -555, -557, -559

Tabelle 388: 4-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Ausgabewert Kanal 1
n+1	D3	D2	Ausgabewert Kanal 2
n+2	D5	D4	Ausgabewert Kanal 3
n+3	D7	D6	Ausgabewert Kanal 4

Diese Ausgangsklemmen stellen sich mit 4x2 Byte dar und belegen in Klasse (0x68) 4 Instanzen.



13.3.5 Sonderklemmen

Bei einzelnen Klemmen wird neben den Datenbytes auch das Control-/ Statusbyte eingeblendet. Dieses dient dem bidirektionalen Datenaustausch der Busklemme mit der übergeordneten Steuerung.

Das Control- bzw. Steuerbyte wird von der Steuerung an die Klemme und das Statusbyte von der Klemme an die Steuerung übertragen. Somit ist beispielsweise das Setzen eines Zählers mit dem Steuerbyte oder die Anzeige von Bereichsunter- oder -überschreitung durch das Statusbyte möglich.

Das Control-/Statusbyte liegt bei dem Feldbuskoppler/-controller mit EtherNet/IP stets im Low-Byte.

Information



Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Internetseite unter: http://www.wago.com.

Die Sonderklemmen stellen sich wie analoge Klemmen dar.
Deshalb belegen deren Prozesseingangswerte pro Kanal ebenfalls eine Instanz in dem Analog Input Point Object (Class 0x67) und deren Prozessausgangswerte pro Kanal eine Instanz in dem Analog Input Point Object (Class 0x68).

13.3.5.1 Zählerklemmen

750-404, (und alle Varianten außer /000-005),
753-404, (und Variante /000-003)

Die Zählerklemmen erscheinen mit insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie ein zusätzliches Steuer-/Statusbyte. Die Busklemmen liefern dann 32-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 389: Zählerklemmen 750-404, (und alle Varianten außer /000-005), 753-404, (und Variante /000-003)

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	S	Statusbyte
	D1	D0	Zählerwert
	D3	D2	

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild			
Instanz	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
n	-	C	Steuerbyte
	D1	D0	Zählersetzwert
	D3	D2	

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

750-404/000-005

Die Zählerklemmen erscheinen mit insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich der Prozessabbilder, 4 Datenbytes sowie ein zusätzliches Steuer-/Statusbyte. Diese Busklemmen liefern pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 390: Zählerklemmen 750-404/000-005

Eingangsprozessabbild			
Instanz	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
n	-	S	Statusbyte
	D1	D0	Zählerwert Zähler 1
	D3	D2	Zählerwert Zähler 2

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild			
Instanz	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
n	-	C	Steuerbyte
	D1	D0	Zählersetzwert Zähler 1
	D3	D2	Zählersetzwert Zähler 2

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x68) 1 Instanz.

750-638,
753-638

Diese Zählerklemmen erscheinen mit insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Die Busklemmen liefern dann pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.



Tabelle 391: Zählerklemmen 750-638, 753-638

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	S0	Statusbyte von Zähler 1
	D1	D0	Zählerwert von Zähler 1
n+1	-	S1	Statusbyte von Zähler 2
	D3	D2	Zählerwert von Zähler 2

Diese Sonderklemmen stellen sich mit 2x3 Byte dar und belegen in Klasse (0x67) 2 Instanzen.

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C0	Steuerbyte von Zähler 1
	D1	D0	Zählersetzwert von Zähler 1
n+1	-	C1	Steuerbyte von Zähler 2
	D3	D2	Zählersetzwert von Zähler 2

Diese Sonderklemmen stellen sich mit 2x3 Byte dar und belegen in Klasse (0x68) 2 Instanzen.

13.3.5.2 Pulsweitenklemmen

750-511, (und alle Varianten /xxx-xxx)

Diese Pulsweitenklemmen erscheinen mit insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 392: Pulsweitenklemmen 750-511, /xxx-xxx

Ein- und Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C0/S0	Steuer-/Statusbyte von Kanal 1
	D1	D0	Datenwert von Kanal 1
n+1			Steuer-/Statusbyte von Kanal 2
	D3	D2	Datenwert von Kanal 2

Diese Sonderklemmen stellen sich jeweils mit 2x3 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

13.3.5.3 Serielle Schnittstellen mit alternativem Datenformat

750-650, (und die Varianten /000-002, -004, -006, -009, -010, -011, -012, -013),
750-651, (und die Varianten /000-002, -003),
750-653, (und die Varianten /000-002, -007),
753-650, -653

Hinweis



Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrierten Betriebart!
Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Abhängig davon, ist das Prozessabbild dieser Busklemmen dann das gleiche, wie das von der entsprechenden Variante.

Die seriellen Schnittstellenklemmen, die mit dem alternativen Datenformat eingestellt sind, erscheinen mit insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer- /Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 393: Serielle Schnittstellen mit alternativem Datenformat

<table>
<tr><th colspan="5">Ein- und Ausgangsprozessabbild</th></tr>
<tr><th rowspan="2">Instanz</th><th colspan="2">Bezeichnung der Bytes</th><th colspan="2" rowspan="2">Bemerkung</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td>n</td><td>D0</td><td>C/S</td><td>Datenbyte</td><td>Steuer- /Statusbyte</td></tr>
<tr><td>n+1</td><td>D2</td><td>D1</td><td colspan="2">Datenbytes</td></tr>
</table>

Diese Sonderklemmen stellen sich jeweils mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

13.3.5.4 Serielle Schnittstellen mit Standard Datenformat

750-650/000-001, -014, -015, -016
750-651/000-001
750-653/000-001, -006

Die seriellen Schnittstellenklemmen, die mit dem Standard Datenformat eingestellt sind, erscheinen mit insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches Steuer- /Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 394: Serielle Schnittstellen mit Standard Datenformat

<table>
<tr><th colspan="5">Ein- und Ausgangsprozessabbild</th></tr>
<tr><th rowspan="2">Instanz</th><th colspan="2">Bezeichnung der Bytes</th><th colspan="2" rowspan="2">Bemerkung</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td rowspan="3">n</td><td>D0</td><td>C/S</td><td>Datenbyte</td><td>Steuer- /Statusbyte</td></tr>
<tr><td>D2</td><td>D1</td><td colspan="2" rowspan="2">Datenbytes</td></tr>
<tr><td>D4</td><td>D3</td></tr>
</table>

Diese Sonderklemmen stellen sich jeweils mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.



13.3.5.5 Datenaustauschklemmen

750-654, (und die Variante /000-001)

Die Datenaustauschklemmen erscheinen mit jeweils insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 395: Datenaustauschklemmen

Ein- und Ausgangsprozessabbild			
Instanz	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
n	D1	D0	Datenbytes
n+1	D3	D2	

Diese Sonderklemmen stellen sich jeweils mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

13.3.5.6 SSI-Geber Interface Busklemmen

750-630, (und die Varianten /000-001, -002, -006, -008, -009, -011, -012, -013)

Hinweis



Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrierten Betriebart!
Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Abhängig davon, ist das Prozessabbild dieser Busklemmen dann das gleiche, wie das von der entsprechenden Variante.

Die SSI-Geber Interface Busklemmen mit Status erscheinen mit insgesamt 4 Datenbytes im Eingangsbereich des Prozessabbilds. Dabei werden mit word-alignment insgesamt 2 Worte im Prozessabbild belegt.

Tabelle 396: SSI-Geber Interface Busklemmen mit alternativem Datenformat

Eingangsprozessabbild			
Instanz	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
n	D1	D0	Datenbytes
n+1	D3	D2	

Diese Sonderklemmen stellen sich mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen.

750-630/000-004, -005, -007

Die SSI-Geber Interface Busklemmen mit Status erscheinen mit insgesamt 5 Bytes Nutzdaten im Eingangsprozessabbild, 4 Datenbytes und ein zusätzliches Statusbyte. Dabei werden mit word-alignment insgesamt 3 Worte im Prozessabbild belegt.

Tabelle 397: SSI-Geber Interface Busklemmen mit alternativem Datenformat

Eingangsprozessabbild				
Instanz	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
n	-	S	nicht genutzt	Statusbyte
	D1	D0	Datenbytes	
	D3	D2		

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

13.3.5.7 Weg- und Winkelmessung

750-631/000-004, -010, -011

Die Busklemme erscheint mit 5 Bytes im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 398: Weg- und Winkelmessung 750-631/000-004, --010, -011

Eingangsprozessabbild				
Instanz	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
n	-	S	nicht genutzt	Statusbyte
	D1	D0	Zählerwort	
	-	-	nicht genutzt	
	D4	D3	Latchwort	

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild			
Instanz	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
n	-	C	Steuerbyte von Zähler 1
	D1	D0	Zählersetzwert von Zähler 1
	-	-	nicht genutzt
	-	-	nicht genutzt

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x68) 1 Instanz.

750-634

Die Busklemme 750-634 erscheint mit 5 Bytes (in der Betriebsart Periodendauermessung mit 6 Bytes) im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.



Tabelle 399: Weg- und Winkelmessung 750-634

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	S	nicht genutzt	Statusbyte
	D1	D0	Zählerwort	
	-	(D2) *)	nicht genutzt	(Periodendauer)
	D4	D3	Latchwort	
*)	Ist durch das Steuerbyte die Betriebsart Periodendauermessung eingestellt, wird in D2 zusammen mit D3/D4 die Periodendauer als 24-Bit-Wert ausgegeben.			

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	C	nicht genutzt	Steuerbyte
	D1	D0	Zählersetzwort	
	-	-	nicht genutzt	
	-	-		

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x68) 1 Instanz.

750-637

Die Inkremental Encoder Interface Busklemme erscheint mit 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes und zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 400: Inkremental Encoder Interface Busklemme

Ein- und Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C0/S0	Steuer-/Statusbyte von Kanal 1
	D1	D0	Datenwerte von Kanal 1
n+1	-	C1/S1	Steuer-/Statusbyte von Kanal 2
	D3	D2	Datenwerte von Kanal 2

Diese Sonderklemmen stellen sich jeweils mit 2x3 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

750-635,
753-635

Die Digitale Impuls Schnittstelle erscheint mit insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	D0	C0/S0	Datenbyte	Steuer-/Statusbyte
	D2	D1	Datenbytes	

Diese Sonderklemmen stellen sich jeweils mit 1x4 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.8 DC-Drive Controller

750-636

Der DC-Drive Controller 750-636 stellt dem Feldbuskoppler/-controller über 1 logischen Kanal 6 Byte Ein- und Ausgangsprozessabbild zur Verfügung. Die zu sendenden und zu empfangenden Positionsdaten werden in 4 Ausgangsbytes (D0 ... D3) und 4 Eingangsbytes (D0 ... D3) abgelegt. 2 Steuerbytes (C0, C1) und 2 Statusbytes (S0, S1) dienen zur Steuerung der Busklemme und des Antriebs. Alternativ zu den Positionsdaten im Eingangsprozessabbild (D0 ... D3) können erweiterte Statusinformationen (S2 ... S5) eingeblendet werden. Die 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen den Prozessdaten und den erweiterten Statusbytes im Eingangsprozessabbild erfolgt über Bit 3 (ExtendedInfo_ON) im Controlbyte C1 (C1.3). Mit Bit 3 des Statusbytes S1 (S1.3) wird die Umschaltung quittiert.

Tabelle 401: Antriebssteuerung 750-636

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	S1	S0	Status S1	Statusbyte S0
	D1*) / S3**)	D0*) / S2**)	Istposition*) / Erweitertes Statusbyte S3**)	Istposition (LSB)*) / Erweitertes Statusbyte S2**)
	D3*) / S5**)	D2*) / S4**)	Istposition (MSB)*) / Erweitertes Statusbyte S3**)	Istposition*) / Erweitertes Statusbyte S4**)
*) ExtendedInfo_ON = ‘0’. **) ExtendedInfo_ON = ‘1’.				

Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	C1	C0	Steuerbyte C1	Steuerbyte C0
	D1	D0	Sollposition	Sollposition (LSB)
	D3	D2	Sollposition (MSB)	Sollposition

Diese Sonderklemmen stellen sich jeweils mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.



13.3.5.9 Steppercontroller

750-670

Der Steppercontroller RS 422 / 24 V / 20 mA 750-670 stellt dem Feldbuskoppler/-controller über 1 logischen Kanal 12 Byte Ein- und Ausgangsprozessabbild zur Verfügung.
Die zu sendenden und zu empfangenden Daten werden in Abhängigkeit von der Betriebsart in bis zu 7 Ausgangsbytes (D0 ... D6) und 7 Eingangsbytes (D0 ... D6) abgelegt. Das Ausgangsbyte D0 und das Eingangsbyte D0 sind reserviert und ohne Funktion. Ein Klemmenbus-Steuer- und Statusbyte (C0, S0) sowie 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen beiden Prozessabbildern erfolgt über das Bit 5 im Controlbyte C0 (C0.5). Mit dem Bit 5 des Statusbytes S0 (S0.5) wird das Einschalten der Mailbox quittiert.

Tabelle 402: Steppercontroller RS 422 / 24 V / 20 mA 750-670

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	Reserviert	S0	Reserviert	Statusbyte S0
	D1	D0	Prozessdaten*) / Mailbox**)	
	D3	D2		
	D5	D4		
	S3	D6	Statusbyte S3	Prozessdaten*) / Reserviert**)
	S1	S2	Statusbyte S1	Statusbyte S2

*) Zyklisches Prozessabbild (Mailbox ausgeschaltet).
**) Mailboxprozessabbild (Mailbox eingeschaltet)

Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	Reserviert	C0	Reserviert	Controlbyte C0
	D1	D0	Prozessdaten*) / Mailbox**)	
	D3	D2		
	D5	D4		
	C3	D6	Controlbyte C3	Prozessdaten*) / Reserviert**)
	C1	C2	Controlbyte C1	Controlbyte C2

*) Zyklisches Prozessabbild (Mailbox ausgeschaltet).
**) Mailboxprozessabbild (Mailbox eingeschaltet)

Diese Sonderklemmen stellen sich jeweils mit 1x12 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.10 RTC-Modul

750-640

Das RTC-Modul erscheint mit insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes, ein zusätzliches Steuer-/Statusbyte und jeweils ein Befehlsbyte (ID). Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 403: RTC-Modul 750-640

Ein- und Ausgangsprozessabbild				
Instanz	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
n	ID	C/S	Befehlsbyte	Steuer-/Statusbyte
	D1	D0	Datenbytes	
	D3	D2		

Diese Sonderklemmen stellen sich jeweils mit 1x 6 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.11 DALI/DSI-Masterklemme

750-641

Die DALI/DSI-Masterklemme erscheint mit insgesamt 6 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 404: DALI/DSI-Masterklemme 750-641

Eingangsprozessabbild				
Instanz	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
n	D0	S	DALI-Antwort	Statusbyte
	D2	D1	Message 3	DALI-Adresse
	D4	D3	Message 1	Message 2

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild				
Instanz	Bezeichnung der Bytes		Bemerkung	
	High Byte	Low Byte		
n	D0	C	DALI-Befehl, DSI-Dimmwert	Steuerbyte
	D2	D1	Parameter 2	DALI-Adresse
	D4	D3	Command-Extension	Parameter 1

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x68) 1 Instanz.



13.3.5.12 Funkreceiver EnOcean

750-642

Die EnOcean Funkreceiverklemme erscheint mit insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Die 3 Bytes Ausgangsdaten werden jedoch nicht genutzt. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 405: Funkreceiver EnOcean 750-642

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	D0	S	Datenbyte	Statusbyte
n+1	D2	D1	Datenbytes	

Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	C	nicht genutzt	Steuerbyte
n+1	-	-	nicht genutzt	

Diese Sonderklemmen stellen sich jeweils mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

13.3.5.13 MP-Bus-Masterklemme

750-643

Die MP-Bus-Masterklemme erscheint mit insgesamt 8 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 6 Datenbytes und zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 406: MP-Bus-Masterklemme 750-643

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	C1/S1	C0/S0	erweitertes Steuer-/Statusbyte	Steuer-/Statusbyte
	D1	D0	Datenbytes	
	D3	D2		
	D5	D4		

Diese Sonderklemmen stellen sich jeweils mit 1x8 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.14 *Bluetooth®* RF-Transceiver

750-644

Die Größe des Prozessabbildes der *Bluetooth®*-Busklemme ist in den festgelegten Größen 12, 24 oder 48 Byte einstellbar.
Es besteht aus einem Steuerbyte (Eingang) bzw. Statusbyte (Ausgang), einem Leerbyte, einer 6, 12 oder 18 Byte großen, überlagerbaren Mailbox (Modus 2) und den *Bluetooth®*-Prozessdaten in einem Umfang von 4 bis 46 Byte.

Die *Bluetooth®*-Busklemme belegt also jeweils 12 bis maximal 48 Bytes im Prozessabbild, wobei die Größen des Eingangs- und Ausgangsprozessabbildes stets übereinstimmen.

Das erste Byte enthält das Steuer-/Statusbyte, das zweite ein Leerbyte.
Daran schließen sich bei ausgeblendeter Mailbox unmittelbar Prozessdaten an.
Bei eingeblendeter Mailbox werden je nach deren Größe die ersten 6, 12 oder 18 Byte Prozessdaten von Mailbox-Daten überlagert. Die Bytes im Bereich hinter der optional einblendbaren Mailbox enthalten grundsätzlich Prozessdaten. Den internen Aufbau der *Bluetooth®*-Prozessdaten entnehmen Sie der Dokumentation des *Bluetooth®* RF-Transceivers 750-644.

Die Einstellung der Mailbox- und Prozessabbildgrößen erfolgt mit dem Inbetriebnahmetool WAGO-I/O-*CHECK*.

Tabelle 407: Bluetooth® RF-Transceiver 750-644

<table>
<tr><th colspan="5">Ein- und Ausgangsprozessabbild</th></tr>
<tr><th rowspan="2">Instanz</th><th colspan="2">Bezeichnung der Bytes</th><th rowspan="2" colspan="2">Bemerkung</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td rowspan="6">n</td><td>-</td><td>C0/S0</td><td>nicht genutzt</td><td>Steuer-/Statusbyte</td></tr>
<tr><td>D1</td><td>D0</td><td rowspan="5" colspan="2">Mailbox (0, 3, 6 oder 9 Worte) sowie Prozessdaten (2-23 Worte)</td></tr>
<tr><td>D3</td><td>D2</td></tr>
<tr><td>D5</td><td>D4</td></tr>
<tr><td>...</td><td>...</td></tr>
<tr><td>D45</td><td>D44</td></tr>
</table>

Die 750-644 stellt sich als Sonderklemme dar. Ihre Prozessdaten (12, 24 oder 48 Byte) belegen je eine Instanz in Klasse 0x67 und 0x68.



13.3.5.15 Schwingstärke/Wälzlagerüberwachung VIB I/O

750-645

Die Schwingstärke/Wälzlagerüberwachung VIB I/O erscheint mit insgesamt 12 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 8 Datenbytes und vier zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 8 Worte im Prozessabbild belegt.

Tabelle 408: Schwingstärke/Wälzlagerüberwachung VIB I/O 750-645

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	C0/S0	nicht genutzt	Steuer-/Statusbyte (log. Kanal 1, Sensoreingang 1)
	D1	D0	Datenbytes (log. Kanal 1, Sensoreingang 1)	
n+1	-	C1/S1	nicht genutzt	Steuer-/Statusbyte (log. Kanal 2, Sensoreingang 2)
	D3	D2	Datenbytes (log. Kanal 2, Sensoreingang 2)	
n+2	-	C2/S2	nicht genutzt	Steuer-/Statusbyte (log. Kanal 3, Sensoreingang 3)
	D5	D4	Datenbytes (log. Kanal 3, Sensoreingang 3)	
n+3	-	C3/S3	nicht genutzt	Steuer-/Statusbyte (log. Kanal 4, Sensoreingang 4)
	D7	D6	Datenbytes (log. Kanal 4, Sensoreingang 4)	

Diese Sonderklemmen stellen sich jeweils mit 4x3 Byte dar und belegen in Klasse (0x67) 4 Instanzen und in Klasse (0x68) 4 Instanzen.

13.3.5.16 AS-interface Masterklemme

750-655

Das Prozessabbild der AS-interface Masterklemme ist in seiner Länge einstellbar in den festgelegten Größen von 12, 20, 24, 32, 40 oder 48 Byte.
Es besteht aus einem Control- bzw. Statusbyte, einer 0, 6, 10, 12 oder 18 Byte großen Mailbox und den AS-interface Prozessdaten in einem Umfang von 0 bis 32 Byte.
Mit word-alignment belegt die AS-interface Masterklemme also jeweils 6 bis maximal 24 Worte im Prozessabbild.

Das erste Ein- bzw. Ausgangswort enthält das Status- bzw. Controlbyte sowie ein Leerbyte.
Daran schließen sich für die fest eingeblendete Mailbox (Modus 1) die Worte mit Mailboxdaten an.
Wenn die Mailbox überlagerbar eingestellt ist (Modus 2), enthalten diese Worte Mailbox- oder Prozessdaten.
Die weiteren Worte enthalten die restlichen Prozessdaten.

Tabelle 409: AS-interface Masterklemme 750-655

<table>
<tr><th colspan="5">Ein- und Ausgangsprozessabbild</th></tr>
<tr><th rowspan="2">Instanz</th><th colspan="2">Bezeichnung der Bytes</th><th colspan="2" rowspan="2">Bemerkung</th></tr>
<tr><th>High Byte</th><th>Low Byte</th></tr>
<tr><td rowspan="6">n</td><td>-</td><td>C0/S0</td><td>nicht genutzt</td><td>Steuer-/Statusbyte</td></tr>
<tr><td>D1</td><td>D0</td><td colspan="2" rowspan="5">Mailbox (0,3,5,6 oder 9 Worte) / Prozessdaten (0-16 Worte)</td></tr>
<tr><td>D3</td><td>D2</td></tr>
<tr><td>D5</td><td>D4</td></tr>
<tr><td>...</td><td>...</td></tr>
<tr><td>D45</td><td>D44</td></tr>
</table>

Diese Sonderklemmen stellen sich jeweils mit 1x 12...48 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.



13.3.6 Systemklemmen

13.3.6.1 Systemklemmen mit Diagnose

750-610, -611

Die Potentialeinspeiseklemmen 750-610 und –611 mit Diagnose liefern zur Überwachung der Versorgung 2 Bits in das Prozesseingangsabbild.

Tabelle 410: Systemklemmen mit Diagnose 750-610, -611

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnosebit S 2 Sicherung	Diagnosebit S 1 Spannung

Die Eingangsklemmen belegen in Klasse (0x65) 2 Instanzen.

13.3.6.2 Binäre Platzhalterklemme

750-622

Die binären Platzhalterklemmen 750-622 verhalten sich wahlweise wie 2 Kanal Digitaleingangsklemmen oder -ausgangsklemmen und belegen je nach angewählter Einstellung pro Kanal 1, 2, 3 oder 4 Bits.

Dabei werden dann entsprechend 2, 4, 6 oder 8 Bits entweder im Prozesseingangs- oder -ausgangsabbild belegt.

Tabelle 411: Binäre Platzhalterklemmen 750-622

Ein- oder Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(Datenbit DI 8)	(Datenbit DI 7)	(Datenbit DI 6)	(Datenbit DI 5)	(Datenbit DI 4)	(Datenbit DI 3)	Datenbit DI 2	Datenbit DI 1

Die Platzhalterklemmen belegen in Klasse (0x65) bzw. in Klasse (0x66) 2, 4, 6 oder 8 Instanzen.

14 Anwendungsbeispiele

14.1 Test von MODBUS-Protokoll und Feldbusknoten

Zum Testen der Funktion Ihres Feldbusknotens benötigen Sie einen MODBUS-Master. Hierfür werden unterschiedliche PC-Applikationen von diversen Herstellern angeboten, die Sie zum Teil als kostenfreie Demoversionen aus dem Internet herunterladen können.

Eines der Programme zum Test Ihres ETHERNET-Feldbusknotens ist **ModScan** der Firma Win-Tech.

Information



Weitere Information
Eine kostenlose Demoversion des Programmes ModScan32 sowie weitere Zusatzprogramme der Firma Win-Tech finden Sie im Internet unter:
http://www.win-tech.com/html/demos.htm

ModScan32 ist eine Windows-Applikation, die als MODBUS-Master arbeitet.

Mit diesem Programm können Sie auf die Datenpunkte Ihres angeschlossenen ETHERNET-TCP/IP-Feldbusknotens zugreifen und gewünschte Änderungen vornehmen.

Information



Weitere Information
Eine Beispiel-Beschreibung zur Software-Bedienung finden Sie im Internet unter: http://www.win-tech.com/html/modscan32.htm

14.2 Visualisierung und Steuerung mittels SCADA-Software

Dieses Kapitel vermittelt Ihnen einen kurzen Einblick zum Einsatz des (programmierbaren) WAGO-ETHERNET-Feldbuskopplers/-controllers mit einer Standard-Anwendersoftware zur Prozessvisualisierung und -steuerung.

Das Angebot an Prozessvisualisierungsprogrammen diverser Hersteller, sogenannte SCADA-Software, ist vielfältig.

Information



Weitere Information
Eine Auswahl an SCADA-Produkten finden Sie z. B. unter:
www.iainsider.co.uk/scadasites.htm

SCADA ist die Abkürzung für „Supervisory Control and Data Acquisition“ und umfasst Fernwirk- und Datenerfassungssysteme.



Dabei handelt es sich um produktionsnahe, bedienerorientierte Werkzeuge, die als Produktionsinformationssysteme für die Bereiche Automatisierungstechnik, Prozesssteuerung und Produktionsüberwachung genutzt werden.

Der Einsatz von SCADA-Systemen umfasst die Bereiche Visualisierung und Überwachung, Datenzugriff, Trendaufzeichnung, Ereignis- und Alarmbearbeitung, Prozessanalyse sowie den gezielten Eingriff in einen Prozess (Steuerung).

Der WAGO-ETHERNET-Feldbusknoten stellt dazu die benötigten Prozesseingangs- und -ausgangswerte bereit.

Hinweis



Nur SCADA-Software mit MODBUS-Unterstützung und MODBUS-Treiber verwenden!
Achten Sie bei der Auswahl einer geeigneten SCADA-Software unbedingt darauf, dass ein MODBUS-Gerätetreiber zur Verfügung steht und die im Feldbuskoppler/-controller realisierten MODBUS/TCP-Funktionen unterstützt werden.

Visualisierungsprogramme mit MODBUS-Gerätetreiber werden u. a. von den Firmen Wonderware, National Instruments, Think&Do oder KEPware Inc. angeboten und sind teilweise auch als Demoversion im Internet frei erhältlich.

Die Bedienung dieser Programme ist herstellerspezifisch. Dennoch sind im Folgenden einige wesentliche Schritte aufgeführt, die veranschaulichen, wie ein Programm mit einem WAGO-ETHERNET-Feldbusknoten und einer SCADA-Software prinzipiell entwickelt werden kann:

1. Laden Sie zunächst den MODBUS-Treiber und wählen Sie MODBUS-ETHERNET.

2. Geben Sie die IP-Adresse zur Adressierung des Feldbusknotens ein.

In einigen Programmen können zudem Aliasnamen, z. B. „Messdaten“, für einen Knoten vergeben werden. Die Adressierung kann dann über diesen Namen erfolgen.

3. Kreieren Sie ein grafisches Objekt, wie beispielsweise einen Schalter (digital) oder ein Potenziometer (analog).

Das kreierte Objekt wird auf der Benutzeroberfläche dargestellt.

4. Verknüpfen Sie das Objekt mit dem gewünschten Datenpunkt an dem Knoten, indem Sie folgende Daten eingeben:

   - Knotenadresse (IP-Adresse oder Aliasnamen)
   - Gewünschter MODBUS-Funktionscode (Register/Bit lesen/schreiben)
   - MODBUS-Adresse des gewählten Kanals

Die Eingabe erfolgt programmspezifisch.

Die MODBUS-Adresse eines Busklemmenkanals enthält je nach Anwendersoftware bis zu 5 Stellen.

Beispiel einer MODBUS-Adressierung

Bei der SCADA-Software Lookout der Firma National Instruments werden 6-stellige MODBUS-Adressen verwendet.
Dabei repräsentiert die erste Stelle die MODBUS-Tabelle (0, 1, 3 oder 4) und implizit den Funktionscode (siehe nachfolgende Tabelle).

Tabelle 412: MODBUS-Tabelle und -Funktionscodes

MODBUS-Tabelle	MODBUS-Funktionscode	
0	FC1 oder FC15	Lesen eines Eingangsbits oder Schreiben mehrerer Ausgangsbits
1	FC2	Lesen mehrerer Eingangsbits
3	FC4	Lesen mehrerer Eingangsregistern
4	FC3 oder FC 16	Lesen mehrerer Eingangsregistern oder Schreiben mehrerer Ausgangsregister

Die folgenden fünf Stellen geben die Kanalnummer (beginnend mit 1) der durchnummerierten digitalen oder analogen Eingangs- oder Ausgangskanäle an.

Beispiele:

- Lesen/Schreiben des ersten digitalen Einganges: z. B. 0 0000 1
- Lesen/Schreiben des zweiten analogen Einganges: z. B. 3 0000 2



Anwendungsbeispiel:

Mit der Eingabe: „Messdaten . 0 0000 2“ kann beispielsweise der digitale Eingangskanal 2 des o. g. Knotens „Messdaten“ ausgelesen werden.

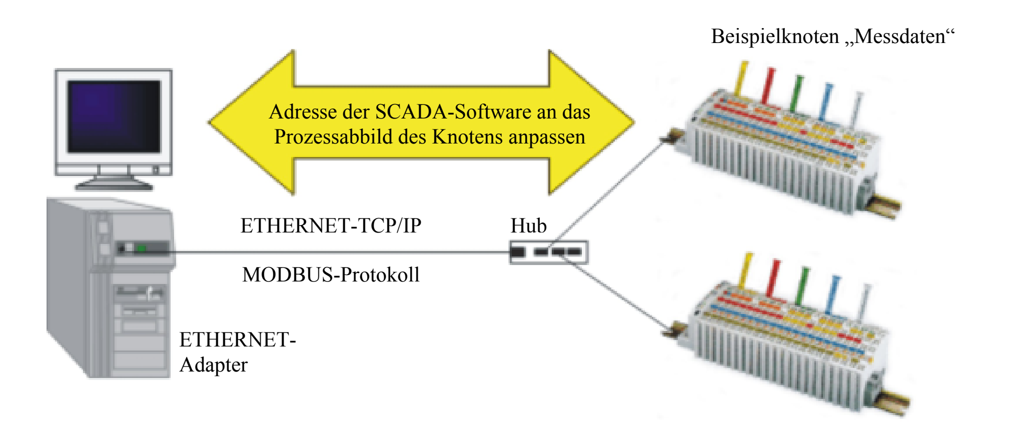


Abbildung 84: Beispiel SCADA-Software mit MODBUS-Treiber

Information



Weitere Information
Eine detaillierte Beschreibung der jeweiligen Software-Bedienung entnehmen Sie dem Handbuch des entsprechenden SCADA-Produktes.

15 Einsatz in explosionsgefährdeten Bereichen

Das **WAGO-I/O-SYSTEM 750** (elektrische Betriebsmittel) ist für den Einsatz in explosionsgefährdeten Bereichen der Zone 2 ausgelegt.

Die nachfolgenden Kapitel beinhalten die allgemeine Kennzeichnung der Komponenten sowie die zu berücksichtigenden Errichtungsbestimmungen. Die einzelnen Abschnitte im Kapitel „Errichtungsbestimmungen“ müssen berücksichtigt werden, falls die Busklemme die entsprechende Zulassung besitzt oder dem Anwendungsbereich der ATEX-Richtlinie unterliegt.



15.1 Kennzeichnung

15.1.1 Für Europa gemäß CENELEC und IEC

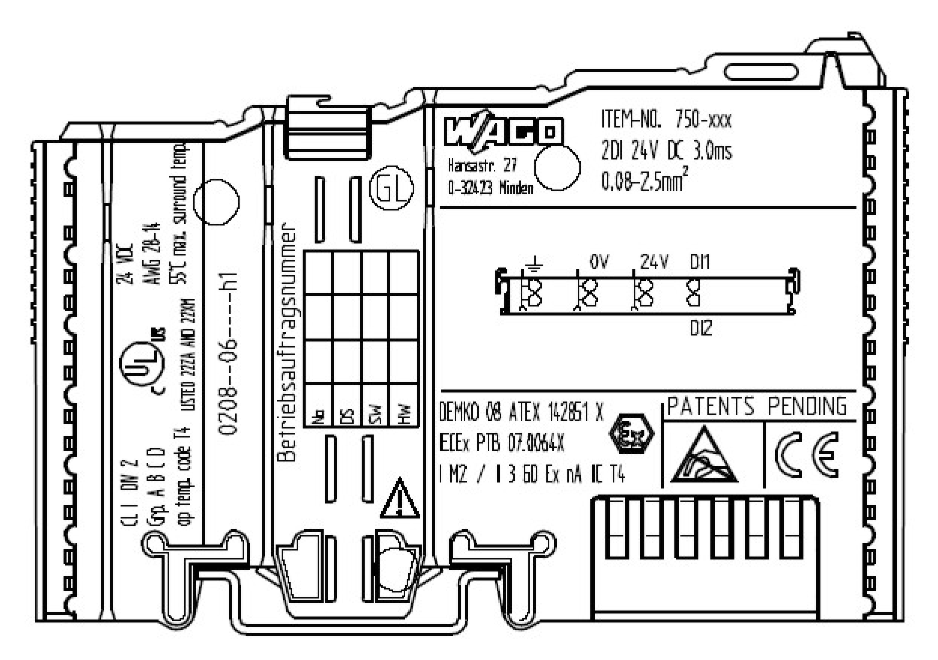


Abbildung 85: Beispiel für seitliche Beschriftung der ATEX- und IEC-Ex- zugelassenen Busklemmen

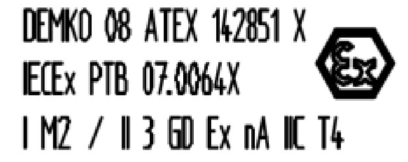


Abbildung 86: Bedruckungstext Detail gemäß CENELEC und IEC

Tabelle 413: Beschreibung der Bedruckung

Bedruckungstext	Beschreibung
DEMKO 08 ATEX 142851 X IECEx PTB 07.0064X	Zulassungsbehörde bzw. Nummer des Untersuchungszertifikats
I M2 / II 3 GD	Explositionsschutzgruppe und Gerätekategorie
Ex nA	Zündschutzart und erweiterte Kennzeichnung
IIC	Explosionsschutzgruppe
T4	Temperaturklasse

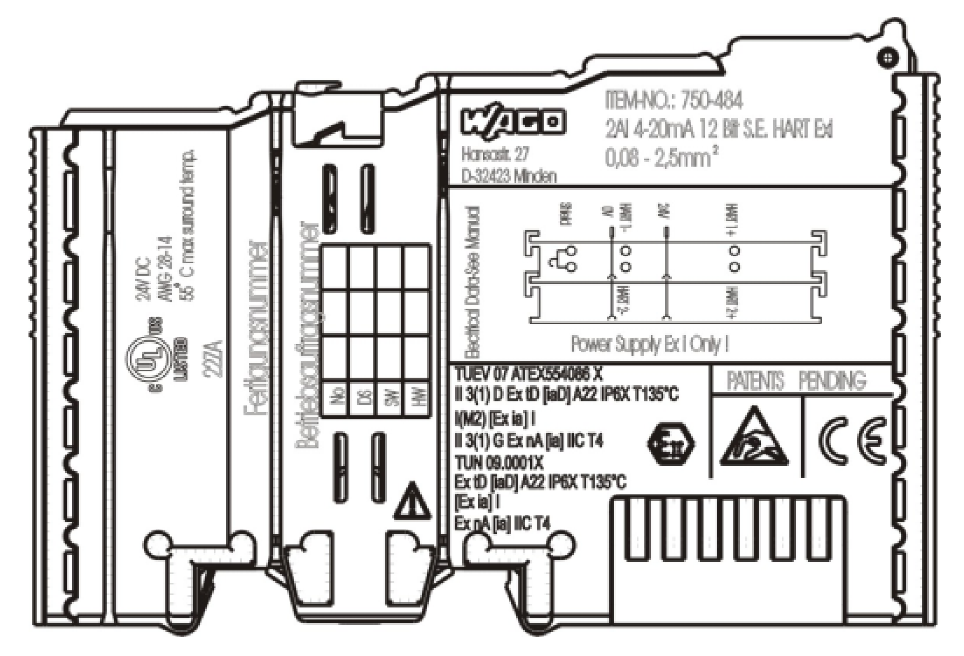


Abbildung 87: Beispiel für seitliche Beschriftung der Ex i und IEC Ex i zugelassenen Busklemmen

TUEV 07 ATEX554086 X
II 3(1) D Ex tD [iaD] A22 IP6X T135°C
I(M2) [Ex ia] I
II 3(1) G Ex nA [ia] IIC T4
TUN 09.0001X
Ex tD [iaD] A22 IP6X T135°C
[Ex ia] I
Ex nA [ia] IIC T4

Abbildung 88: Bedruckungstext Detail gemäß CENELEC und IEC

Tabelle 414: Beschreibung der Bedruckung

Bedruckungstext	Beschreibung
TÜV 07 ATEX 554086 X TUN 09.0001X	Zulassungsbehörde bzw. Bescheinigungsnummern
Stäube	
II	Gerätegruppe: alle außer Bergbau
3(1)D	Gerätekategorie: Zone 22 Gerät (Zone 20 Teilgerät)
Ex	Explosionsschutzkennzeichen
tD	Schutz durch Gehäuse
[iaD]	Zugelassen entsprechend Norm “Staub-Eigensicherkeit“
A22	Oberflächentemperatur bestimmt nach Verfahren A, Verwendung in Zone 22
IP6X	Schutz gegen Eindringen von Staub
T 135°C	Max. Oberflächentemp. des Gehäuses (ohne Staubablage)
Bergbau	
I	Gerätegruppe: Bergbau
(M2)	Gerätekategorie: hohes Maß an Sicherheit
[Ex ia]	Explosionsschutz Kennzeichen mit Kategorie der Zündschutzart Eigensicherheit: sicher auch bei auftreten von zwei Fehlern
I	Gerätegruppe: Bergbau
Gase	
II	Gerätegruppe: Alle außer Bergbau
3(1)G	Gerätekategorie: Zone 2 Gerät (Zone 0 Teilgerät)
Ex	Explosionsschutzkennzeichen
nA	Zündschutzart: Nicht Funken gebendes Betriebsmittel
[ia]	Kategorie der Zündschutzart Eigensicherheit: Sicher auch bei auftreten von zwei Fehlern
IIC	Explosionsgruppe
T4	Temperaturklasse: Max. Oberflächentemperatur 135°C

15.1.2 Für Amerika gemäß NEC 500

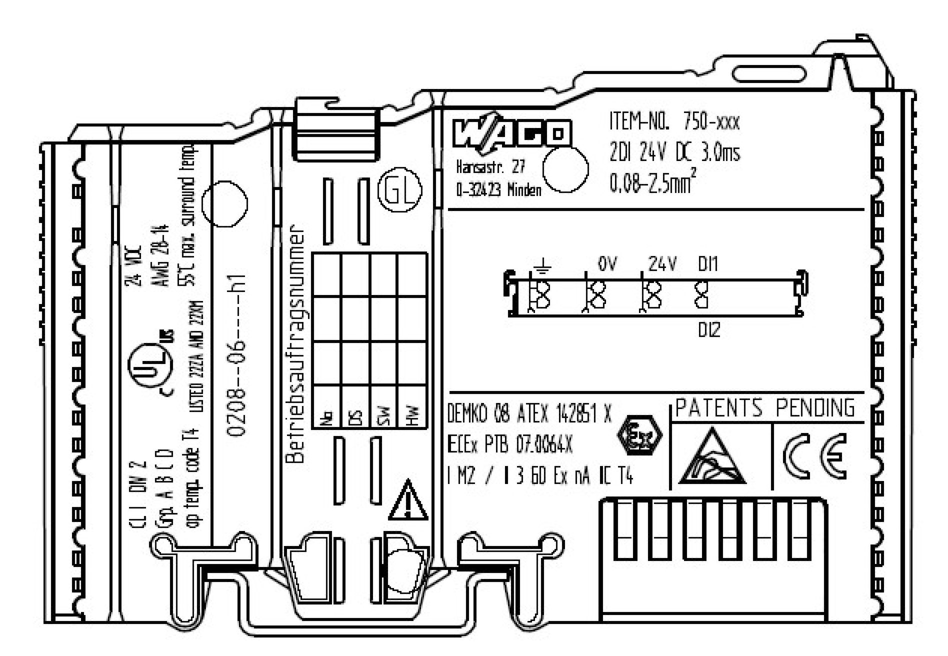


Abbildung 89: Beispiel für seitliche Beschriftung der Busklemmen

CL I DIV 2
Grp. A B C D
op temp. code T4
LISTED 22ZA AND 22XM

Abbildung 90: Bedruckungstext Detail gemäß NEC

Tabelle 415: Beschreibung der Bedruckung

Bedruckungstext	Beschreibung
CL 1	Explosionsschutzgruppe (Gefahrenkategorie)
DIV 2	Einsatzbereich (Zone)
Grp. ABCD	Explosionsgruppe (Gasgruppe)
Optemp code T4	Temperaturklasse

15.2 Errichtungsbestimmungen

In der **Bundesrepublik Deutschland** sind verschiedene nationale Bestimmungen und Verordnungen für das Errichten von elektrischen Anlagen in explosionsgefährdeten Bereichen zu beachten. Die Grundlage hierfür bildet die Betriebssicherheitsverordnung, welche die nationale Umsetzung der europäischen Richtlinie 99/92/E6 ist. Ihr zugeordnet ist die Errichtungsbestimmung EN 60079-14. Nachfolgend sind auszugsweise zusätzliche VDE-Bestimmungen zu finden:

Tabelle 416: VDE-Errichtungsbestimmungen in Deutschland

DIN VDE 0100	Errichten von Starkstromanlagen mit Nennspannungen bis 1000 V
DIN VDE 0101	Errichten von Starkstromanlagen mit Nennspannungen über 1 kV
DIN VDE 0800	Errichtung und Betrieb von Fernmeldeanlagen einschließlich Informationsverarbeitungsanlagen
DIN VDE 0185	Blitzschutzanlagen

In den **USA** und **Kanada** gelten eigenständige Vorschriften. Nachfolgend sind auszugsweise diese Bestimmungen aufgeführt:

Tabelle 417: Errichtungsbestimmungen in USA und Kanada

NFPA 70	National Electrical Code Art. 500 Hazardous Locations
ANSI/ISA-RP 12.6-1987	Recommended Practice
C22.1	Canadian Electrical Code

ACHTUNG



Nachfolgende Punkte beachten!
Der Einsatz des **WAGO-I/O-SYSTEMs 750** (elektrisches Betriebsmittel) mit Ex-Zulassung erfordert unbedingt die Beachtung nachfolgender Punkte für die entsprechenden Einsatzbereiche:

15.2.1 Besondere Bedingungen für den sicheren ATEX- und IEC-Ex-Betrieb gem. DEMKO 08 ATEX 142851X und IECEx PTB 07.0064

Die feldbusunabhängigen Busklemmen des WAGO-I/O-SYSTEMs 750-.../...-... müssen in einer Umgebung mit Verschmutzungsgrad 2 oder besser installiert werden. In der Endanwendung sind die Busklemmen in einem Gehäuse mit mindestens der Schutzart IP54 einzusetzen mit folgenden Ausnahmen:

- Die Busklemmen 750-440, 750-609 und 750-611 müssen in einem Gehäuse mit mindestens der Schutzart IP64 eingebaut werden.
- Die Busklemme 750-540 muss für 230 V AC Anwendungen in einem Gehäuse mit mindestens der Schutzart IP64 eingebaut werden.
- Die Busklemme 750-440 darf nur maximal bis 120 V AC eingesetzt werden.

Bei Anwendungen, in denen eine Gefährdung durch brennbare Stäube auftreten kann, müssen alle Geräte und das Gehäuse gemäß den Anforderungen der IEC 61241-0:2006 und IEC 61241-1:2004 vollständig getestet und beurteilt werden.

Bei Anwendungen im Bergbau müssen alle Geräte gemäß den Anforderungen der EN 60079-0:2006 und EN 60079-1:2002 installiert und als Betriebsmittel zertifiziert werden.

Das Installieren, Hinzufügen, Entfernen oder Ersetzen von Busklemmen, Feldbussteckern oder Sicherungen darf nur erfolgen, wenn die System- und Feldversorgung ausgeschaltet sind, oder der Bereich keine explosionsfähige Atmosphäre aufweist.

DIP-Schalter, Kodierschalter und Potentiometer, die an die Busklemme angeschlossen sind, dürfen nur betätigt werden, wenn explosionsfähige Atmosphäre ausgeschlossen werden kann.

Die Busklemme 750-642 darf nur in Verbindung mit der Antennen 758-910 mit einer max. Kabellänge von 2,5 m eingesetzt werden.

Um die Bemessungsspannung nicht mehr als 40 % zu überschreiten, ist ein Transientenschutz an den Versorgungsanschlüssen vorzusehen.

Der zulässige Umgebungstemperaturbereich beträgt 0 °C bis +55 °C.



15.2.2 Besondere Bedingungen für den sicheren Ex Betrieb (ATEX Zertifikat TÜV 07 ATEX 554086 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGO-I/O-SYSTEM 750-*** in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach der Richtlinie 94/9/EG und der zutreffenden Normen (siehe Kennzeichnung) EN 60079-0, EN 60079-11, EN 60079-15, EN 61241-0 und EN 61241-1 erfüllt. Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß EN 60079-0 und EN 60079-1 und einen IP-Gehäuseschutz von IP64 gewährleistet. Die Übereinstimmung mit diesen Anforderungen und dem korrekten Einbau des Gerätes in ein Gehäuse oder Schaltschrank muss durch einen ExNB bescheinigt sein.

2. Werden die Schnittstellenstromkreise ohne die Feldbuskoppelstation des Typs 750-3../...-... (DEMKO 08 ATEX 142851 X) betrieben, so sind außerhalb des Gerätes Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40% überschritten wird.

3. DIP-Schalter, Kodierschalter und Potentiometer, die an die Busklemme angeschlossen sind, dürfen nur betätigt werden, wenn explosionsfähige Atmosphäre ausgeschlossen werden kann.

4. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.

5. Für die Typen 750-606, 750-625/000-001, 750-487/003-000, 750-484 muss Folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der IEC 60664-1 definiert, begrenzt werden.

6. Für den Typ 750-601 ist Folgendes zu berücksichtigen: Die Sicherung darf nicht entfernt oder getauscht werden, wenn das Gerät in Betrieb ist.

7. Der zulässige Umgebungstemperaturbereich beträgt $0°C \leq T_a \leq +55°C$ (erweiterte Angaben entnehmen Sie bitte dem Zertifikat).

15.2.3 Besondere Bedingungen für den sicheren Ex Betrieb (IEC-Ex Zertifikat TUN 09.0001 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGO-I/O-SYSTEM 750-*** in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach den zutreffenden Normen (siehe Kennzeichnung) IEC 60079-0, IEC 60079-11, IEC 60079-15, IEC 61241-0 und IEC 61241-1 erfüllt. Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß IEC 60079-0 und IEC 60079-1 mit der Schutzart IP64, gewährleistet. Eine Konformitätserklärung muss die Übereinstimmung dieser Anforderungen und den korrekten Einbau der Geräte im Gehäuse oder Schaltschrank durch eine Ex-Zertifizierungsstelle bestätigen.

2. Außerhalb des Gerätes sind Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40% überschritten wird.

3. DIP-Schalter, Kodierschalter und Potentiometer, die an die Busklemme angeschlossen sind, dürfen nur betätigt werden, wenn explosionsfähige Atmosphäre ausgeschlossen werden kann.

4. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.

5. Für die Typen 750-606, 750-625/000-001, 750-487/003-000, 750-484 muss Folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der IEC 60664-1 definiert, begrenzt werden.

6. Für den Typ 750-601 ist Folgendes zu berücksichtigen: Die Sicherung darf nicht entfernt oder getauscht werden, wenn das Gerät in Betrieb ist.

7. Der zulässige Umgebungstemperaturbereich beträgt $0°C \leq T_a \leq +55°C$ (erweiterte Angaben entnehmen Sie bitte dem Zertifikat).



15.2.4 ANSI/ISA 12.12.01

Dieses Gerät ist ausschließlich für den Einsatz in Klasse I, Division 2, Gruppen A, B, C, D oder nicht explosionsgefährdeten Bereichen geeignet.

ACHTUNG



Explosionsgefahr!
Explosionsgefahr - der Austausch von Komponenten kann die Eignung für Klasse I, Div. 2 beeinträchtigen.

ACHTUNG



Gerät nur stromfrei und in nicht-explosionsgefährdeten Bereichen abklemmen!
Klemmen Sie das Gerät nur dann ab, wenn der Strom ausgeschaltet ist oder wenn der Bereich in der Nähe jedes Bedieners, der zugänglichen Steckers und Sicherungshalter als nicht-explosionsgefährdet gilt.

Für Baugruppen mit Sicherungen muss folgender Hinweis angebracht werden:
„Es muss ein Schalter vorgesehen sein, der für den Einsatzort geeignet ist, wo das Gerät installiert wird, um die Sicherung spannungsfrei zu schalten“.
Der Schalter muss nicht in dem Gerät eingebaut sein.

Für Baugruppen mit ETHERNET-Steckern:
„Nur für den Einsatz in LAN, nicht für den Anschluss an Fernmeldeleitungen“.

ACHTUNG



Nur mit Antennenmodul 758-910 verwenden!
Benutzen Sie die Klemme 750-642 nur mit einem Antennenmodul 758-910.

Information



Weitere Information
Einen Zertifizierungsnachweis erhalten Sie auf Anfrage. Beachten Sie auch die Hinweise auf dem Beipackzettel der Busklemme. Das Handbuch mit den oben aufgeführten Bedingungen für sicheren Gebrauch muss für den Anwender zu jederzeit zur Verfügung stehen.

16 Anhang

16.1 MIB-II-Gruppen

16.1.1 System Group

Die System Group enthält allgemeine Informationen zum Feldbuskoppler/-controller.

Tabelle 418: MIB II – System Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.2.1.1.1	sysDescr	R	Der Eintrag enthält die Geräteidentifikation. Der Eintrag wird fest z. B. auf "WAGO 750-841" codiert.
1.3.6.1.2.1.1.2	sysObjectID	R	Der Eintrag enthält die Autorisierungs-Identifikation des Herstellers.
1.3.6.1.2.1.1.3	sysUpTime	R	Der Eintrag enthält die Zeit in hundertstel Sekunden seit dem letzten zurücksetzen der Management Einheit.
1.3.6.1.2.1.1.4	sysContakt	R/W	Der Eintrag enthält die Identifikation der Kontaktperson und enthält Informationen wie diese zu erreichen ist.
1.3.6.1.2.1.1.5	sysName	R/W	Dieser Eintrag enthält einen Administrativen Namen für das Gerät.
1.3.6.1.2.1.1.6	sysLocation	R/W	Dieser Eintrag enthält den physikalischen Einbauort des Knotens
1.3.6.1.2.1.1.7	sysServices	R	Dieser Eintrag bezeichnet die Menge von Diensten, welche dieser Feldbuskoppler/-controller enthält.



16.1.2 Interface Group

Die Interface Group enthält Informationen und Statistiken zu dem Geräteinterface.

Ein Geräteinterface beschreibt die ETHERNET-Schnittstelle des Kopplers/Controllers und liefert die Statusinformationen der physikalischen ETHERNET-Ports sowie der internen Loopback-Schnittstelle.

Tabelle 419: MIB II – Interface Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.2.1.2.1	ifNumber	R	Anzahl der Netzwerkschnittstellen in diesem System
1.3.6.1.2.1.2.2	ifTable	-	Liste der Netzwerkschnittstellen
1.3.6.1.2.1.2.2.1	ifEntry	-	Eintrag der Netzwerkschnittstelle
1.3.6.1.2.1.2.2.1.1	ifIndex	R	Eindeutige Zuordnungsnummer jeder Schnittstelle
1.3.6.1.2.1.2.2.1.2	ifDescr	R	Name des Herstellers, Produktname und Version der Hardware-Schnittstelle, z. B. „WAGO Kontakttechnik GmbH 750-841: Rev 1.0“
1.3.6.1.2.1.2.2.1.3	ifType	R	Typ der Schnittstelle: ETHERNET-CSMA/CD = 6 Software-Loopback = 24
1.3.6.1.2.1.2.2.1.4	ifMtu	R	Maximale Telegrammlänge (Maximal-Transfer-Unit), die über diese Schnittstelle transferiert werden kann
1.3.6.1.2.1.2.2.1.5	ifSpeed	R	Geschwindigkeit der Schnittstelle in Bit/s an
1.3.6.1.2.1.2.2.1.6	ifPhysAddress	R	Physikalische Adresse der Schnittstelle (im Fall von ETHERNET, die MAC-Adresse)
1.3.6.1.2.1.2.2.1.7	ifAdmin-Status	R/W	Gewünschter Zustand der Schnittstelle Mögliche Werte: up(1): Betriebsbereit zum Senden und Empfangen down(2): Schnittstelle ist abgeschaltet testing(3): Schnittstelle befindet sich im Testmodus
1.3.6.1.2.1.2.2.1.8.	ifOperStatus	R	Gegenwärtiger Zustand der Schnittstelle Dieser Parameter hat keine Relevanz für Port 1 und Port 2.
1.3.6.1.2.1.2.2.1.9.	ifLastChange	R	Wert von sysUpTime; Zeitpunkt, in dem sich der Zustand zum letzten Mal geändert hat
1.3.6.1.2.1.2.2.1.10	ifInOctets	R	Anzahl aller über die Schnittstelle empfangenen Daten in Bytes
1.3.6.1.2.1.2.2.1.11	ifInUcastPkts	R	Anzahl der empfangenen Unicast-Pakete, die an eine höhere Schicht weitergeleitet wurden
1.3.6.1.2.1.2.2.1.12	ifInNUcastPkts	R	Anzahl der empfangenen Broadcast- und Multicast-Pakete, die an eine höhere Schicht weitergeleitet wurden
1.3.6.1.2.1.2.2.1.13	ifInDiscards	R	Anzahl der Pakete, die vernichtet worden sind, obwohl keine Störungen vorliegen
1.3.6.1.2.1.2.2.1.14	ifInErrors	R	Anzahl der eingegangenen fehlerhaften Pakete, die nicht an eine höhere Schicht weitergeleitet worden sind

Tabelle 419: MIB II – Interface Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.2.1.2.2.1.15	IfInUnknown-Protos	R	Anzahl der eingegangenen Pakete, die an eine nicht bekannte oder nicht unterstützte Port-nummer gesendet wurden.
1.3.6.1.2.1.2.2.1.16	ifOutOctets	R	Anzahl aller bisher über die Schnittstelle gesendeten Daten in Bytes
1.3.6.1.2.1.2.2.1.17	ifOutUcastPkts	R	Anzahl der gesendeten Unicast-Pakete, die an eine höhere Schicht weitergeleitet wurden
1.3.6.1.2.1.2.2.1.18	ifOutNUcastPkts	R	Anzahl der gesendeten Broadcast- und Multicast-Pakete, die an eine höhere Schicht weitergeleitet wurden
1.3.6.1.2.1.2.2.1.19	ifOutDiscards	R	Anzahl der Pakete, die vernichtet worden sind, obwohl keine Störungen vorliegen
1.3.6.1.2.1.2.2.1.20	ifOutErrors	R	Anzahl, der aufgrund von Fehlern, nicht versendeten Pakete

16.1.3 IP Group

Die IP-Group enthält Informationen über die IP-Vermittlung.

Tabelle 420: MIB II – IP Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.2.1.4.1	ipForwarding	R/W	1: Host ist Router; 2: Host ist kein Router
1.3.6.1.2.1.4.2	ipDefaultTTL	R/W	Default-Wert für das Time-To-Live-Feld jedes IP-Frames
1.3.6.1.2.1.4.3	ipInReceives	R	Anzahl der empfangenen IP-Frames einschließlich der fehlerhaften Frames
1.3.6.1.2.1.4.4	ipInHdrErrors	R	Anzahl der empfangenen IP-Frames mit Headerfehlern
.3.6.1.2.1.4.5	ipInAddrErrors	R	Anzahl der empfangenen IP-Frames mit fehlgeleiteter IP-Adresse
1.3.6.1.2.1.4.6	ipForwDatagrams	R	Anzahl der empfangenen IP-Frames die weitergeleitet (geroutet) wurden
1.3.6.1.2.1.4.7	ipUnknownProtos	R	Anzahl der empfangenen IP-Frames mit einem unbekannten Protokolltyp
1.3.6.1.2.1.4.8	ipInDiscards	R	Anzahl der empfangenen IP-Frames ohne Fehler, die trotzdem verworfen wurden
1.3.6.1.2.1.4.9	ipInDelivers	R	Anzahl der empfangenen IP-Frames die an höhere Protokollschichten weitergeleitet wurden
1.3.6.1.2.1.4.10	ipOutRequests	R	Anzahl der gesendeten IP-Frames
1.3.6.1.2.1.4.11	ipOutDiscards	R	Anzahl der zu sendenden, jedoch verworfenen IP-Frames
1.3.6.1.2.1.4.12	ipOutNoRoutes	R	Anzahl gesendeter und wegen fehlerhafter Routing-Informationen verworfener IP-Frames
1.3.6.1.2.1.4.13	ipReasmTimeout	R	Mindestzeitdauer bis ein IP-Frame wieder zusammengesetzt wird
1.3.6.1.2.1.4.14	ipReasmReqds	R	Mindestanzahl der IP-Fragmente zum Zusammensetzen und Weiterleiten
1.3.6.1.2.1.4.15	ipReasmOKs	R	Anzahl der erfolgreich wieder zusammengesetzten IP-Frames
1.3.6.1.2.1.4.16	ipReasmFails	R	Anzahl der nicht erfolgreich wieder zusammengesetzten IP-Frames
1.3.6.1.2.1.4.17	ipFragOKs	R	Anzahl der IP-Frames, die fragmentiert und weitergeleitet wurden
1.3.6.1.2.1.4.18	ipFragFails	R	Anzahl der zu fragmentierenden IP-Frames, die aufgrund des „don't-fragment-bits“, das im Header gesetzt ist, nicht fragmentiert werden
1.3.6.1.2.1.4.19	ipFragCreates	R	Anzahl der erzeugten IP-Fragment-Frames
1.3.6.1.2.1.4.20	ipAddrTable	-	Tabelle aller lokalen IP-Adressen des Gerätes
1.3.6.1.2.1.4.20.1	ipAddrEntry	-	Adressinformationen für einen Eintrag
1.3.6.1.2.1.4.20.1.1	ipAdEntAddr	R	Die IP-Adresse betreffenden Adressinformationen
1.3.6.1.2.1.4.20.1.2	ipAdEntIfIndex	R	Index der Schnittstelle
1.3.6.1.2.1.4.20.1.3	ipAdEntNetMask	R	Die zugehörige Subnetzmaske zu dem Eintrag
1.3.6.1.2.1.4.20.1.4	ipAdEntBcastAddr	R	Wert des niederwertigsten Bits in der IP-Broadcast-Adresse
1.3.6.1.2.1.4.20.1.5	IpAdEntReasm-MaxSize	R	Die Größe des längsten IP-Telegramms, das wieder defragmentiert werden kann
1.3.6.1.2.1.4.23	ipRoutingDiscards	R	Anzahl der gelöschten Routing-Einträge

16.1.4 IpRoute Table Group

Die IP-RouteTable enthält Informationen über die Routing-Tabelle in dem Feldbuskoppler/-controller.

Tabelle 421: MIB II – IpRoute Table Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.2.1.4.21	ipRouteTable	-	IP-Routing-Tabelle
1.3.6.1.2.1.4.21.1	ipRouteEntry	-	Ein Routing-Eintrag für ein bestimmtes Ziel
1.3.6.1.2.1.4.21.1.1	ipRouteDest	R/W	Dieser Eintrag gibt die Zieladresse des Routing-Eintrags an
1.3.6.1.2.1.4.21.1.2	ipRouteIfIndex	R/W	Dieser Eintrag gibt den Index des Interfaces an, welches das nächste Ziel der Route ist
1.3.6.1.2.1.4.21.1.3	ipRouteMetric1	R/W	Die primäre Route zum Zielsystem
1.3.6.1.2.1.4.21.1.4	ipRouteMetric2	R/W	Eine alternative Route zum Zielsystem
1.3.6.1.2.1.4.21.1.5	ipRouteMetric3	R/W	Eine alternative Route zum Zielsystem
1.3.6.1.2.1.4.21.1.6	ipRouteMetric4	R/W	Eine alternative Route zum Zielsystem
.3.6.1.2.1.4.21.1.7	ipRouteNextHop	R/W	Die IP-Addresse des nächsten Teilstücks der Route
1.3.6.1.2.1.4.21.1.8	ipRouteType	R/W	Die Art der Route
1.3.6.1.2.1.4.21.1.9	ipRouteProto	R	Mechanismus wie die Route aufgebaut wird
1.3.6.1.2.1.4.21.1.10	ipRouteAge	R/W	Anzahl der Sekunden, seitdem die Route das letzte mal erneuert wurde oder überprüft wurde
1.3.6.1.2.1.4.21.1.11	ipRouteMask	R/W	Der Eintrag enthält die Subnetmask zu diesem Eintrag
1.3.6.1.2.1.4.21.1.12	ipRouteMetric5	R/W	Eine alternative Route zum Zielsystem
1.3.6.1.2.1.4.21.1.13	ipRouteInfo	R/W	Ein Verweis auf eine spezielle MIB



16.1.5 ICMP Group

Tabelle 422: MIB II – ICMP Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.2.1.5.1	icmpInMsgs	R	Anzahl der empfangenen ICMP-Meldungen
1.3.6.1.2.1.5.2	icmpInErrors	R	Anzahl der empfangenen ICMP-Meldungen, die ICMP-spezifische Fehler enthalten
1.3.6.1.2.1.5.3	icmpInDestUnreachs	R	Anzahl der empfangenen ICMP-Destination-Unreachable-Meldungen
1.3.6.1.2.1.5.4	icmpInTimeExcds	R	Anzahl der empfangenen ICMP-Time-Exceeded-Meldungen
1.3.6.1.2.1.5.5	icmpInParmProbs	R	Anzahl der empfangenen ICMP-Parameterproblemmeldungen
1.3.6.1.2.1.5.6	icmpInSrcQuenchs	R	Anzahl der empfangenen ICMP-Source-Quench-Meldungen
1.3.6.1.2.1.5.7	icmpInRedirects	R	Anzahl der empfangenen ICMP-Redirect-Meldungen
1.3.6.1.2.1.5.8	icmpInEchos	R	Anzahl der empfangenen ICMP-Echo-Request-Meldungen (Ping)
1.3.6.1.2.1.5.9	icmpInEchoReps	R	Anzahl der empfangenen ICMP-Echo-Reply-Meldungen (Ping)
1.3.6.1.2.1.5.10	icmpInTimestamps	R	Anzahl der empfangenen ICMP-Timestamp-Request-Meldungen
1.3.6.1.2.1.5.11	icmpInTimestampReps	R	Anzahl der empfangenen ICMP-Timestamp Reply-Meldungen
1.3.6.1.2.1.5.12	icmpInAddrMasks	R	Anzahl der empfangenen ICMP-Address-Mask-Request-Meldungen
1.3.6.1.2.1.5.13	icmpInAddrMaskReps	R	Anzahl der empfangenen ICMP-Address-Mask-Reply-Meldungen
1.3.6.1.2.1.5.14	icmpOutMsgs	R	Anzahl der gesendeten ICMP-Meldungen
1.3.6.1.2.1.5.15	icmpOutErrors	R	Anzahl gesendeter ICMP-Meldungen, die wegen Problemen nicht gesendet werden konnten
1.3.6.1.2.1.5.16	icmpOutDestUnreachs	R	Anzahl der gesendeten ICMP-Destination-Unreachable-Meldungen
1.3.6.1.2.1.5.17	icmpOutTimeExcds	R	Anzahl der gesendeten ICMP-Time-Exceeded-Meldungen
1.3.6.1.2.1.5.18	icmpOutParmProbs	R	Anzahl der gesendeten ICMP-Parameterproblemmeldungen
1.3.6.1.2.1.5.19	icmpOutSrcQuenchs	R	Anzahl der gesendeten ICMP-Source-Quench-Meldungen
1.3.6.1.2.1.5.20	icmpOutRedirects	R	Anzahl der gesendeten ICMP-Redirection-Meldungen
1.3.6.1.2.1.5.21	icmpOutEchos	R	Anzahl der gesendeten ICMP-Echo-Request-Meldungen
1.3.6.1.2.1.5.22	icmpOutEchoReps	R	Anzahl der gesendeten ICMP-Echo-Reply-Meldungen
1.3.6.1.2.1.5.23	icmpOutTimestamps	R	Anzahl der gesendeten ICMP-Timestamp-Request-Meldungen
1.3.6.1.2.1.5.24	icmpOutTimestampReps	R	Anzahl der gesendeten ICMP-Timestamp-Reply-Meldungen
1.3.6.1.2.1.5.25	icmpOutAddrMasks	R	Anzahl der gesendeten ICMP-Address-Mask-Request-Meldungen
1.3.6.1.2.1.5.26	icmpOutAddrMaskReps	R	Anzahl der gesendeten ICMP-Address-Mask-Reply-Meldungen

16.1.6 TCP Group

Tabelle 423: MIB II – TCP Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.2.1.6.1	tcpRtoAlgorithm	R	Retransmission-time (1 = andere, 2 = konstant, 3 = MIL-Standart 1778, 4 = Jacobson)
1.3.6.1.2.1.6.2	tcpRtoMin	R	Minimaler Wert für den Retransmission-Timer
1.3.6.1.2.1.6.3	tcpRtoMax	R	Maximaler Wert für den Retransmission-Timer
1.3.6.1.2.1.6.4	tcpMaxConn	R	Anzahl maximaler TCP-Verbindungen, die gleichzeitig bestehen können
1.3.6.1.2.1.6.5	tcpActiveOpens	R	Anzahl der bestehenden aktiven TCP-Verbindungen
1.3.6.1.2.1.6.6	tcpPassiveOpens	R	Anzahl der bestehenden passiven TCP-Verbindungen
1.3.6.1.2.1.6.7	tcpAttemptFails	R	Anzahl der fehlgeschlagenen Verbindungsaufbauversuche
1.3.6.1.2.1.6.8	tcpEstabResets	R	Anzahl der Verbindungsneustarts
1.3.6.1.2.1.6.9	tcpCurrEstab	R	Anzahl der TCP-Verbindungen im Established- oder Close-Wait-Zustand
1.3.6.1.2.1.6.10	tcpInSegs	R	Anzahl der empfangenen TCP-Frames einschließlich der Error-Frames
1.3.6.1.2.1.6.11	tcpOutSegs	R	Anzahl der korrekt gesendeten TCP-Frames mit Daten
1.3.6.1.2.1.6.12	tcpRetransSegs	R	Anzahl der gesendeten TCP-Frames die wegen Fehlern wiederholt wurden
1.3.6.1.2.1.6.13	tcpConnTable	-	Für jede bestehende Verbindung wird ein Tabelleneintrag erzeugt
1.3.6.1.2.1.6.13.1	tcpConnEntry	-	Tabelleneintrag zur Verbindung
1.3.6.1.2.1.6.13.1.1	tcpConnState	R	Status der TCP-Verbindung
1.3.6.1.2.1.6.13.1.2	tcpConnLocalAddress	R	IP-Adresse für diese Verbindung (bei Servern fest eingestellt auf 0.0.0.0)
1.3.6.1.2.1.6.13.1.3	tcpConnLocalPort	R	Portnummer der TCP-Verbindung
1.3.6.1.2.1.6.13.1.4	tcpConnRemAddress	R	Remote IP-Adresse der TCP-Verbindung
1.3.6.1.2.1.6.13.1.5	tcpConnRemPort	R	Remote-Port der TCP-Verbindung
1.3.6.1.2.1.6.14	tcpInErrs	R	Anzahl der empfangenen fehlerhaften TCP-Frames
1.3.6.1.2.1.6.15	tcpOutRsts	R	Anzahl der gesendeten TCP-Frames mit gesetztem RST-Flag



16.1.7 UDP Group

Tabelle 424: MIB II – UDP Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.2.1.7.1	udpInDatagrams	R	Anzahl empfangener UDP-Frames, die an die entsprechenden Applikationen weitergegeben wurden
1.3.6.1.2.1.7.2	udpNoPorts	R	Anzahl empfangener UDP-Frames, die nicht an die entsprechenden Applikationen weiter-gegeben werden konnten (port unreachable)
1.3.6.1.2.1.7.3	udpInErrors	R	Anzahl empfangener UDP-Frames, die aus anderen Gründen nicht weitergegeben werden konnten
1.3.6.1.2.1.7.4	udpOutDatagrams	R	Anzahl gesendeter UDP-Frames
1.3.6.1.2.1.7.5	udpTable	-	Für jede Applikation die UDP-Frames erhalten hat, wird ein Tabelleneintrag erzeugt
1.3.6.1.2.1.7.5.1	udpEntry	-	Tabelleneintrag für eine Applikation, die einen UDP-Frame erhalten hat
1.3.6.1.2.1.7.5.1.1	udpLocalAddress	R	IP-Adresse des lokalen UDP-Server
1.3.6.1.2.1.7.5.1.2	udpLocalPort	R	Portnummer des lokalen UDP-Server

16.1.8 SNMP Group

Tabelle 425: MIB II – SNMP Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.2.1.11.1	snmpInPkts	R	Anzahl empfangener SNMP-Frames
1.3.6.1.2.1.11.2	snmpOutPkts	R	Anzahl gesendeter SNMP-Frames
1.3.6.1.2.1.11.3	snmpInBadVersions	R	Anzahl empfangener SNMP-Frames mit einer ungültigen Versionsnummer
1.3.6.1.2.1.11.4	snmpInBadCommunity-Names	R	Anzahl empfangener SNMP-Frames mit einer ungültigen community
1.3.6.1.2.1.11.5	snmpInBadCommunity Uses	R	Anzahl empfangener SNMP-Frames, deren community keine ausreichende Berechtigung für die durchzuführenden Aktionen hatten
1.3.6.1.2.1.11.6	snmpInASNParseErrs	R	Anzahl empfangener SNMP-Frames, die einen falschen Aufbau hatten
1.3.6.1.2.1.11.8	snmpInTooBigs	R	Anzahl empfangener SNMP-Frames, die das Ergebnis „tooBig“ zurückmeldeten
1.3.6.1.2.1.11.9	snmpInNoSuchNames	R	Anzahl empfangener SNMP-Frames, die das Ergebnis „noSuchName“ zurückmeldeten
1.3.6.1.2.1.11.10	snmpInBadValues	R	Anzahl empfangener SNMP-Frames, die das Ergebnis „badValue“ zurückmeldeten
1.3.6.1.2.1.11.11	snmpInReadOnlys	R	Anzahl empfangener SNMP-Frames, die das Ergebnis „readOnly“ zurückmeldeten
1.3.6.1.2.1.11.12	snmpInGenErrs	R	Anzahl empfangener SNMP-Frames, die das Ergebnis „genError“ zurückmeldeten
1.3.6.1.2.1.11.13	snmpInTotalReqVars	R	Anzahl empfangener SNMP-Frames mit gültigen GET- oder GET-NEXT-Anforderungen
1.3.6.1.2.1.11.14	snmpInTotalSetVars	R	Anzahl empfangener SNMP-Frames mit gültigen SET-Anforderungen
1.3.6.1.2.1.11.15	snmpInGetRequests	R	Anzahl empfangener und ausgeführter GET-Anforderungen
1.3.6.1.2.1.11.16	snmpInGetNexts	R	Anzahl empfangener und ausgeführter GET-NEXT-Anforderungen
1.3.6.1.2.1.11.17	snmpInSetRequests	R	Anzahl empfangener und ausgeführter SET-Anforderungen
1.3.6.1.2.1.11.18	snmpInGetResponses	R	Anzahl empfangener GET-Antworten
1.3.6.1.2.1.11.19	snmpInTraps	R	Anzahl empfangener Traps
1.3.6.1.2.1.11.20	snmpOutTooBigs	R	Anzahl gesendeter SNMP-Frames, die das Ergebnis „too Big“ enthielten
1.3.6.1.2.1.11.21	snmpOutNoSuchNames	R	Anzahl gesendeter SNMP-Frames, die das Ergebnis „noSuchName“ enthielten
1.3.6.1.2.1.11.22	snmpOutBadValues	R	Anzahl gesendeter SNMP-Frames, die das Ergebnis „badValue“ enthielten
1.3.6.1.2.1.11.24	SnmpOutGenErrs	R	Anzahl gesendeter SNMP-Frames, die das Ergebnis „genErrs“ enthielten
1.3.6.1.2.1.11.25	snmpOutGetRequests	R	Anzahl gesendeter GET-Anforderungen
1.3.6.1.2.1.11.26	SnmpOutGetNexts	R	Anzahl gesendeter GET-NEXT-Anforderungen
1.3.6.1.2.1.11.27	snmpOutSetRequests	R	Anzahl gesendeter SET-Anforderungen
1.3.6.1.2.1.11.28	snmpOutGetResponses	R	Anzahl gesendeter GET-Antworten
1.3.6.1.2.1.11.29	snmpOutTraps	R	Anzahl gesendeter Traps
1.3.6.1.2.1.11.30	snmpEnableAuthenTraps	R/W	Authentification-failure-Traps (1 = ein, 2 = aus)



16.2 WAGO-MIB-Gruppen

16.2.1 Company Group

Die „Company Group” enthält allgemeine Informationen über die Firma WAGO Kontakttechnik GmbH & Co. KG.

Tabelle 426: WAGO-MIB – Company Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.1.1	wagoName	R	Registrierter Firmenname Standardwert: „WAGO Kontakttechnik GmbH & Co. KG“
1.3.6.1.4.1.13576.1.2	wagoDescrition	R	Beschreibung der Firma Standardwert: „WAGO Kontakttechnik GmbH & Co. KG, Hansastr. 27, D-32423 Minden“
1.3.6.1.4.1.13576.1.3	wagoURL	R	"URL for company web site" Defaultwert: "www.wago.com"

16.2.2 Product Group

Die „Produkt Group” enthält Informationen über den Controller.

Tabelle 427: WAGO-MIB – Product Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.1	wioArticleName	R	Artikelname Standardwert: 750-8xx/000-000"
1.3.6.1.4.1.13576.10.1.2	wioArticleDescription	R	Artikelbeschreibung Standardwert: „WAGO Ethernet(10/100MBit)-FBC“
1.3.6.1.4.1.13576.10.1.3	wioSerialNumber	R	Seriennummer des Artikels Standardwert: „SNxxxxxxxx-Txxxxxx-mac|0030DExxxxxx“
1.3.6.1.4.1.13576.10.1.4	wioMacAddress	R	MAC-Adresse des Artikels Standardwert: „0030DExxxxxx“
1.3.6.1.4.1.13576.10.1.5	wioURLDatasheet	R	URL zum Datenblatt des Artikels Standardwert: „http://www.wago.com/ wagoweb/documentation/navigate/nm0dc__e.htm#ethernet“
1.3.6.1.4.1.13576.10.1.6	wioURLManual	R	URL zum Handbuch des Artikels Standardwert: „http://www.wago.com/ wagoweb/documentation/navigate/nm0dc__e.htm#ethernet“

16.2.3 Versions Group

Die „Version Group“ enthält über die verwendeten Hard-/Softwareversionen im Controller.

Tabelle 428: WAGO-MIB – Versions Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.10.1	wioFirmwareIndex	R	Index der Firmeware-Version
1.3.6.1.4.1.13576.10.1.10.2	wioHardwareIndex	R	Index der Hardware-Version
1.3.6.1.4.1.13576.10.1.10.3	wioFwlIndex	R	Index der Software-Version des Firmware-Loaders
1.3.6.1.4.1.13576.10.1.10.4	wioFirmwareVersion	R	Kompletter Firmeware-String



16.2.4 Real-Time Clock Group

Die „Real-Time Clock Group“ enthält Informationen über die Echtzeituhr im System.

Tabelle 429: WAGO-MIB – Real Time Clock Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.11.1	wioRtcDateTime	R/W	Datum/Zeit des Gerätes in UTC-Format als String. Zum Schreiben von Datum/Zeit verwenden Sie folgenden String: „time 11:22:33 date 13-1-2007“ Standardwert: „time xx:xx:xx date xx-xx-xxxx (UTC)“
1.3.6.1.4.1.13576.10.1.11.2	wioRtcTime	R/W	Datum/Zeit des Gerätes in UTC-Format als Integer in Sekunden ab 1970-01-01 Standardwert: „0”
1.3.6.1.4.1.13576.10.1.11.3	wioTimezone	R/W	Aktuelle Zeitzone des Gerätes in Stunden (-12…+12) Standardwert: „0”
1.3.6.1.4.1.13576.10.1.11.4	wioRtcHourMode	R	Stundenmodi: 0 = 12h-Modus 1 = 24h-Modus“ Standardwert: „0”
1.3.6.1.4.1.13576.10.1.11.5	wioRtcBatteryStatus	R	RTC-Batteriestatus: 0 = ok 1 = Batterie leer Standardwert: „1”
1.3.6.1.4.1.13576.10.1.11.6	wioRtcDayLightSaving	R/W	Zeit-Offset von 1 Stunde: 0 = Kein Offset 1 = 1 Stunde Offset (DayLightSaving) Standardwert: „0”

16.2.5 Ethernet Group

Die „Ethernet Group“ enthält die Einstellungen des Controllers für ETHERNET.

Tabelle 430: WAGO-MIB – Ethernet Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.12.1	wioEthernetMode	R/W	IP-Konfiguration der ETHERNET-Verbindung: 0 = feste IP-Adresse 1 = dynamische IP-Adresse über Bootp 2 = dynamische IP-Adresse über DHCP Standardwert: „1”
1.3.6.1.4.1.13576.10.1.12.2	wioIp	R/W	Aktuelle IP-Adresse des Gerätes
1.3.6.1.4.1.13576.10.1.12.3	wioSubnetMask	R/W	Aktuelle Subnetzmaske des Gerätes
1.3.6.1.4.1.13576.10.1.12.4	wioGateway	R/W	Aktuelle Gateway-IP des Gerätes
1.3.6.1.4.1.13576.10.1.12.5	wioHostname	R/W	Aktueller Hostname des Gerätes
1.3.6.1.4.1.13576.10.1.12.6	wioDomainName	R/W	Aktueller Domain-Name des Gerätes
1.3.6.1.4.1.13576.10.1.12.7	wioDnsServer1	R/W	IP-Adresse des 1. DNS-Servers
1.3.6.1.4.1.13576.10.1.12.8	wioDnsServer2	R/W	IP-Adresse des 2. DNS-Servers

16.2.6 Actual Error Group

Die „Actual Error Group“ enthält Informationen zum letzten Systemstatus/Fehlerstatus.

Tabelle 431: WAGO-MIB – Actual Error Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.20.1	wioErrorNumber	R	Fehlernummer des letzten Fehlers
1.3.6.1.4.1.13576.10.1.20.2	wioErrorArgument	R	Fehlerargument des letzten Fehlers
1.3.6.1.4.1.13576.10.1.20.3	wioErrorTime	R	Zeitpunkt des letzten Fehlers

16.2.7 Error History Group

Die „Error History Group“ enthält die letzten System-/Fehlerstatus in einer Tabelle.

Tabelle 432: WAGO-MIB – Error History Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.21.1	wioErrorTable	-	
1.3.6.1.4.1.13576.10.1.21.1.1	wioErrorEntry	-	
1.3.6.1.4.1.13576.10.1.21.1.1.1	wioErrorTableIndex	R	Index der Fehlertabelle
1.3.6.1.4.1.13576.10.1.21.1.1.2	wioErrorTableNumber	R	Tabelle der Fehlernummern
1.3.6.1.4.1.13576.10.1.21.1.1.3	wioErrorTableArgument	R	Tabelle der Fehlerargumente
1.3.6.1.4.1.13576.10.1.21.1.1.4	wioErrorTableTime	R	Tabelle der Fehlerzeitpunkte



16.2.8 PLC Project Group

Die „PLC Project Group“ enthält Informationen des im Controller verwendeten PLC-Programms.

Tabelle 433: WAGO-MIB – PLC Project Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.30.1	wioProjectId	R	ID des CoDeSys-Projektes
1.3.6.1.4.1.13576.10.1.30.2	wioProjectDate	R	Datum des CoDeSys-Projektes
1.3.6.1.4.1.13576.10.1.30.3	wioProjectName	R	Name des CoDeSys-Projektes
1.3.6.1.4.1.13576.10.1.30.4	wioProjectTitle	R	Titel des CoDeSys-Projektes
1.3.6.1.4.1.13576.10.1.30.5	wioProjectVersion	R	Version des CoDeSys-Projektes
1.3.6.1.4.1.13576.10.1.30.6	wioProjectAuthor	R	Autor des CoDeSys-Projektes
1.3.6.1.4.1.13576.10.1.30.7	wioProjectDescription	R	Beschreibung des CoDeSys-Projektes
1.3.6.1.4.1.13576.10.1.30.8	wioNumberOfIecTasks	R	Nummer der IEC-Task des CoDeSys-Projektes
1.3.6.1.4.1.13576.10.1.30.9	wioIecTaskTable	-	
1.3.6.1.4.1.13576.10.1.30.9.1	wioIecTaskEntry	-	
1.3.6.1.4.1.13576.10.1.30.9.1.1	wioIecTaskId	R	ID der IEC-Task im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.2	wioIecTaskName	R	Name der IEC-Task im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.3	wioIecTaskStatus	R	Status der IEC-Task im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.4	wioIecTaskMode	R	Modus der IEC-Task im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.5	wioIecTaskPriority	R	Priorität der IEC-Task im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.6	wioIecTaskInterval	R	Intervall der zyklischen IEC-Tasks im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.7	wioIecTaskEvent	R	Event für IEC-Task im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.8	wioIecTaskCycleCount	R	Zähler für IEC-Tasks im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.9	wioIecTaskCycleTime	R	Letzte Zykluszeit der IEC-Task im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.10	wioIecTaskCycleTime-Min	R	Minimale Zykluszeit der IEC-Task im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.11	wioIecTaskCycleTime-Max	R	Maximale Zykluszeit der IEC-Task im CoDeSys-Projekt
1.3.6.1.4.1.13576.10.1.30.9.1.12	wioIecTaskCycleTime-Avg	R	Durchschnittliche Zykluszeit der IEC-Task im CoDeSys-Projekt

16.2.9 Http Group

Die „Http Group“ enthält Informationen und Einstellungen zum Webserver des Controllers.

Tabelle 434: WAGO-MIB – Http Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.1.1	wioHttpEnable	R/W	Aktivieren/Deaktivieren des Webserver-Ports: 0 = Webserver-Port deaktiviert 1 = Webserver-Port aktiviert Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.1.2	wioHttpAuthen-ticationEnable	R/W	Aktivieren/Deaktivieren der Authentifizierung auf den Internetseiten: 0 = Authentifizierung deaktiviert 1 = Authentifizierung aktiviert Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.1.3	wioHttpPort	R/W	Port des HTTP-Webservers Standardwert: { 80 }

16.2.10 Ftp Group

Die „Ftp Group“ enthält Informationen und Einstellungen zum FTP-Server des Controllers.

Tabelle 435: WAGO-MIB – Ftp Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.2.1	wioFtpEnable	R/W	Aktivieren/Deaktivieren des FTP-Server-Ports: 0 = Port für FTP-Server deaktiviert 1 = Port für FTP-Server aktiviert Standardwert: { 1 }



16.2.11 Sntp Group

Die „Sntp Group“ enthält Informationen und Einstellungen zum SNTP-Server des Controllers.

Tabelle 436: WAGO-MIB – Sntp Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.3.1	wioSntpEnable	R/W	Aktivieren/Deaktivieren des SNTP-Server-Ports: 0 = Port für SNTP-Server deaktiviert 1 = Port für SNTP-Server aktiviert Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.3.2	wioSntpServer-Address	R/W	IP-Adresse des SNTP-Servers Standardwert: { 0 }
1.3.6.1.4.1.13576.10.1.40.3.3	wioSntpClient-Intervall	R/W	Intervall zum Abfragen des SNTP-Managers Standardwert: { 0 }
1.3.6.1.4.1.13576.10.1.40.3.4	wioSntpClient-Timeout	R/W	Timeout zur Unterbrechung der SNTP-Antwort Standardwert: { 2000 }
1.3.6.1.4.1.13576.10.1.40.3.5	wioSntpClient-DayLightSaving	R/W	Zeit-Offset von 1 Stunde: 0 = Kein Offset 1 = 1 Stunde Offset (DayLightSaving) Standardwert: „0”

16.2.12 Snmp Group

Die „Snmp Group“ enthält Informationen und Einstellungen zum SNMP-Agent des Controllers.

Tabelle 437: WAGO-MIB – Snmp Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.4.1	wioSnmpEnable	R/W	Aktivieren/Deaktivieren des SNMP-Server-Ports: 0 = Port für SNMP-Server deaktiviert 1 = Port für SNMP-Server aktiviert Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.4.2.1	wioSnmp1-ProtocolEnable	R/W	Aktivieren/Deaktivieren des 1. SNMPv1/v2c-Agent Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.4.2.2	wioSnmp1-ManagerIp	R/W	IP-Adresse des 1. SNMP-Servers Standardwert: { 'C0A80101'h }
1.3.6.1.4.1.13576.10.1.40.4.2.3	wioSnmp1-Community	R/W	String zur Identifizierung der Community für SNMPv1/v2c Standardwert: { „public“ }
1.3.6.1.4.1.13576.10.1.40.4.2.4	wioSnmp1Trap-V1enable	R/W	Aktivieren/Deaktivieren von SNMPv1-Traps für 1. SNMP-Server Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.4.2.5	wioSnmp1Trap-V2enable	R/W	Aktivieren/Deaktivieren von SNMPv2-Traps für 1. SNMP-Server Standardwert: { 0 }

Tabelle 437: WAGO-MIB – Snmp Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.4.2.6	wioSnmp2-ProtocolEnable	R/W	Aktivieren/Deaktivieren des 1. SNMPv1/v2c-Agent Standardwert: { 0 }
1.3.6.1.4.1.13576.10.1.40.4.2.7	wioSnmp2-ManagerIp	R/W	IP-Adresse des 2. SNMP-Servers Standardwert: { '00000000' h }
1.3.6.1.4.1.13576.10.1.40.4.2.8	wioSnmp2-Community	R/W	String zur Identifizierung der Community für SNMPv1/v2c Standardwert: { „public“ }
1.3.6.1.4.1.13576.10.1.40.4.2.9	wioSnmp2Trap-V1enable	R/W	Aktivieren/Deaktivieren von SNMPv1-Traps für den 1. SNMP-Server Standardwert: { 0 }
1.3.6.1.4.1.13576.10.1.40.4.2.10	wioSnmp2Trap-V2enable	R/W	Aktivieren/Deaktivieren von SNMPv2c-Traps für den 1. SNMP-Server Standardwert: { 0 }
1.3.6.1.4.1.13576.10.1.40.4.3.1	wioSnmp1User-Enable	R/W	Aktivieren/Deaktivieren des 1. SNMPv3- Benutzers Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.4.3.2	wioSnmp1-Authentication-Typ	R/W	Authentifizierungstyp für 1. SNMPv3-Benutzer: 0 = Keine Authentifizierung 1 = MD5-Authentifizierung 2 = SHA1-Authentifizierung Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.4.3.3	wioSnmp1-Authentication-Name	R/W	Authentifizierungsname für 1. SNMPv3-Benutzer Standardwert: { „SecurityName“ }
1.3.6.1.4.1.13576.10.1.40.4.3.4	wioSnmp1-Authentication-Key	R/W	Authentifizierungsschlüssel für 1. SNMPv3-Benutzer Standardwert: { „ AuthenticationKey „ }
1.3.6.1.4.1.13576.10.1.40.4.3.5	wioSnmp1-PrivacyEnable	R/W	Aktivieren/Deaktivieren der Datenverschlüsselung für den 1. SNMPv3-Benutzer: 0 = Keine Verschlüsselung 1 = DES-Verschlüsselung Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.4.3.6	wioSnmp1-PrivacyKey	R/W	Privater Schlüssel für SNMPv3 für den 1. SNMPv3-Benutzer Standardwert: { „PrivacyKey“ }
1.3.6.1.4.1.13576.10.1.40.4.3.7	wioSnmp1-Notification-Enable	R/W	Aktivieren/Deaktivieren von Meldungen (SNMPv3-Traps) mit SNMPv3-Benutzer Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.4.3.8	wioSnmp1-Notification-ReceiverIP	R/W	Empfänder-IP-Adresse für Meldungen (SNMPv3-Traps) mit SNMPv3-Benutzer Standardwert: { 'C0A80101'h }
1.3.6.1.4.1.13576.10.1.40.4.3.9	wioSnmp2User-Enable	R/W	Aktivieren/Deaktivieren des 2. SNMPv3- Benutzers Standardwert: { 0 }
1.3.6.1.4.1.13576.10.1.40.4.3.10	wioSnmp2-Authentication-Typ	R/W	Authentifizierungstyp für 2. SNMPv3-Benutzer: 0 = Keine Authentifizierung 1 = MD5-Authentifizierung



Tabelle 437: WAGO-MIB – Snmp Group

Identifier	Eintrag	Zugriff	Beschreibung
			2 = SHA1-Authentifizierung Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.4.3.1 1	wioSnmp2-Authentication-Name	R/W	Authentifizierungsname für 2. SNMPv3-Benutzer Standardwert: { „SecurityName“ }
1.3.6.1.4.1.13576.10.1.40.4.3.1 2	wioSnmp2-Authentication-Key	R/W	Authentifizierungsschlüssel für 2. SNMPv3-Benutzer Standardwert: { „ AuthenticationKey „ }
1.3.6.1.4.1.13576.10.1.40.4.3.1 3	wioSnmp2-PrivacyEnable	R/W	Privater Schlüssel für SNMPv3 für den 2. SNMPv3-Benutzer Standardwert: { „PrivacyKey“ }
1.3.6.1.4.1.13576.10.1.40.4.3.1 4	wioSnmp2-PrivacyKey	R/W	Privater Schlüssel für SNMPv3 für den 2. SNMPv3-Benutzer Standardwert: { „PrivacyKey“ }
1.3.6.1.4.1.13576.10.1.40.4.3.1 5	wioSnmp2-Notification-Enable	R/W	Aktivieren/Deaktivieren von Meldungen (SNMPv3-Traps) mit SNMPv3-Benutzer Standardwert: { 0 }
1.3.6.1.4.1.13576.10.1.40.4.3.1 6	wioSnmp2-Notification-ReceiverIP	R/W	Empfänder-IP-Adresse für Meldungen (SNMPv3-Traps) mit SNMPv3-Benutzer Standardwert: { '00000000'h }

16.2.13 Snmp Trap String Group

Die „Snmp Trap String Group“ enthält Zeichenketten (Strings), welche an die herstellerspezifischen Traps angehängt werden.

Tabelle 438: WAGO-MIB – Snmp Trap String Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.4.4.1	wioTrapKbus-Error	R/W	Zeichenkette für 1. SNMP-Trap Standardwert: { „Kbus Error“ }
1.3.6.1.4.1.13576.10.1.40.4.4.2	wioTrapPlcStart	R/W	Zeichenkette für 2. SNMP-Trap Standardwert: { „Plc Start“ }
1.3.6.1.4.1.13576.10.1.40.4.4.3	wioTrapPlcStop	R/W	Zeichenkette für 3. SNMP-Trap Standardwert: { „Plc Stop“ }
1.3.6.1.4.1.13576.10.1.40.4.4.4	wioTrapPlc-Reset	R/W	Zeichenkette für 4. SNMP-Trap Standardwert: { „Plc Reset“ }
1.3.6.1.4.1.13576.10.1.40.4.4.5	wioTrapPlcSoft wareWatchdog	R/W	Zeichenkette für 5. SNMP-Trap Standardwert: { „Plc Software Watchdog“ }
1.3.6.1.4.1.13576.10.1.40.4.4.6	wioTrapPlc-DivideByZero	R/W	Zeichenkette für 6. SNMP-Trap Standardwert: { „Plc Divide By Zero“ }
1.3.6.1.4.1.13576.10.1.40.4.4.7	wioTrapPlc-OnlineChange	R/W	Zeichenkette für 7. SNMP-Trap Standardwert: { „Plc Online Change“ }
1.3.6.1.4.1.13576.10.1.40.4.4.8	wioTrapPlc-Download	R/W	Zeichenkette für 8. SNMP-Trap Standardwert: { „Plc Download Programm“ }
1.3.6.1.4.1.13576.10.1.40.4.4.9	wioTrapPlc-Login	R/W	„Zeichenkette für 9. SNMP-Trap Standardwert: { „Plc Login“ }
1.3.6.1.4.1.13576.10.1.40.4.4.10	wioTrapPlc-Logout	R/W	Zeichenkette für 10. SNMP-Trap Standardwert: { „Plc Logout“ }



16.2.14 Snmp User Trap String Group

Die „Snmp User Trap String Group“ enthält Strings, welche an die benutzerspezifischen Traps angehängt werden können. Diese Strings können sowohl über SNMP als auch über die Wago_SNMP.lib im Codesys verändert werden.

Tabelle 439: WAGO-MIB – Snmp User Trap String Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.4.5.1	wioUserTrapMsg1	R/W	Zeichenkette für 1. SNMP-Trap
1.3.6.1.4.1.13576.10.1.40.4.5.2	wioUserTrapMsg2	R/W	Zeichenkette für 2. SNMP-Trap
1.3.6.1.4.1.13576.10.1.40.4.5.3	wioUserTrapMsg3	R/W	Zeichenkette für 3. SNMP-Trap
1.3.6.1.4.1.13576.10.1.40.4.5.4	wioUserTrapMsg4	R/W	Zeichenkette für 4. SNMP-Trap
1.3.6.1.4.1.13576.10.1.40.4.5.5	wioUserTrapMsg5	R/W	Zeichenkette für 5. SNMP-Trap
1.3.6.1.4.1.13576.10.1.40.4.5.6	wioUserTrapMsg6	R/W	Zeichenkette für 6. SNMP-Trap
1.3.6.1.4.1.13576.10.1.40.4.5.7	wioUserTrapMsg7	R/W	Zeichenkette für 7. SNMP-Trap
1.3.6.1.4.1.13576.10.1.40.4.5.8	wioUserTrapMsg8	R/W	Zeichenkette für 8. SNMP-Trap
1.3.6.1.4.1.13576.10.1.40.4.5.9	wioUserTrapMsg9	R/W	Zeichenkette für 9. SNMP-Trap
1.3.6.1.4.1.13576.10.1.40.4.5.10	wioUserTrapMsg10	R/W	Zeichenkette für 10. SNMP-Trap

16.2.15 Plc Connection Group

Über die „Plc Connection Group“ kann die Verbindung zu CoDeSys aktiviert/deaktiviert werden.

Tabelle 440: WAGO-MIB – Plc Connection Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.5.1	wioCoDeSysEnable	R/W	Aktivieren/Deaktivieren des CoDeSys-Server-Ports: 0 = Port für CoDeSys-Server deaktiviert 1 = Port für CoDeSys-Server aktiviert Standardwert: { 1 }

16.2.16 Modbus Group

Die „Modbus Group“ enthält Informationen und Einstellungen zum MODBUS-Server des Controllers.

Tabelle 441: WAGO-MIB – Modbus Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.6.1	wioModbusTcp-Enable	R/W	Aktivieren/Deaktivieren des Modbus-TCP-Server-Ports: 0 = Port für Modbus-TCP-Server deaktiviert 1 = Port für Modbus-TCP-Server aktiviert Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.6.2	wioModbusUdb-Enable	R/W	Aktivieren/Deaktivieren des Modbus-UDP-Server-Ports: 0 = Port für Modbus- UDP -Server deaktiviert 1 = Port für Modbus- UDP -Server aktiviert Standardwert: { 1 }
1.3.6.1.4.1.13576.10.1.40.6.3	wioMax-Connections	R/W	Die maximale Anzahl von MODBUS-Verbindungen Standardwert: { 15 }
1.3.6.1.4.1.13576.10.1.40.6.4	wioConnection-Timeout	R/W	Timeout der MODBUS-Verbindung Standardwert: { 600 }
1.3.6.1.4.1.13576.10.1.40.6.5	wioModbus-WatchdogMode	R/W	Modus des MODBUS-Watchdogs Standardwert: { 0 }
1.3.6.1.4.1.13576.10.1.40.6.6	wioModbus-WatchdogTime	R/W	Timeout des MODBUS-Watchdogs Standardwert: { 100 }
1.3.6.1.4.1.13576.10.1.40.6.7	wioFreeModbus-Sockets	R/W	Ungenutzte und freie MODBUS-Verbindung Standardwert: { 15 }
1.3.6.1.4.1.13576.10.1.40.6.8	wioModbus-ConnectionTable	-	
1.3.6.1.4.1.13576.10.1.40.6.8.1	wioModbus-ConnectionEntry	-	
1.3.6.1.4.1.13576.10.1.40.6.8.1.1	wioModbus-ConnectionIndex	R/W	Index der MODBUS-Verbindung
1.3.6.1.4.1.13576.10.1.40.6.8.1.2	wioModbus-ConnectionIp	R/W	ID-Adresse der MODBUS-Verbindung
1.3.6.1.4.1.13576.10.1.40.6.8.1.3	wioModbus-ConnectionPort	R/W	Port der MODBUS-Verbindung



16.2.17 Ethernet IP Group

Die „Ethernet IP Group“ enthält EtherNet/IP-Informationen und -Einstellungen des Controllers.

Tabelle 442: WAGO-MIB – Ethernet IP Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.40.7.1	wioEthernetIpEnable	R/W	Aktivieren/Deaktivieren des Ethernet/IP-Server-Ports: 0 = Port für Ethernet/IP-Server deaktiviert 1 = Port für Ethernet/IP-Server aktiviert Standardwert: { 0 }
1.3.6.1.4.1.13576.10.1.40.7.2	wioEthernetIpVariables-InputCount	R/W	
1.3.6.1.4.1.13576.10.1.40.7.3	wioEthernetIpVariables-OutputCount	R/W	
1.3.6.1.4.1.13576.10.1.40.7.4	wioEthernetIpVariables-PlcInputCount	R/W	
1.3.6.1.4.1.13576.10.1.40.7.5	wioEthernetIpVariables-PlcInputOffset	R/W	
1.3.6.1.4.1.13576.10.1.40.7.6	wioEthernetIpVariables-PlcOutputCount	R/W	
1.3.6.1.4.1.13576.10.1.40.7.7	wioEthernetIpVariables-PlcOutputOffset	R/W	
1.3.6.1.4.1.13576.10.1.40.7.8	wioEthernetIpRunIdle-HeaderOrginatorToTarget	R/W	
1.3.6.1.4.1.13576.10.1.40.7.9	wioEthernetIpRunIdle-HeaderTargetToOrginator	R/W	

16.2.18 Process Image Group

Die „Process Image Group“ enthält in eine Liste von Informationen über die an den Controller angeschlossenen Klemmen.

Tabelle 443: WAGO-MIB – Process Image Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.50.1	wioModulCount	R	Modulzähler
1.3.6.1.4.1.13576.10.1.50.2	wioAnalogOutLength	R	Länge der Analogausgangsprozessdaten
1.3.6.1.4.1.13576.10.1.50.3	wioAnalogInLength	R	Länge der Analogeingangsprozessdaten
1.3.6.1.4.1.13576.10.1.50.4	wioDigitalOutLength	R	Länge der Digitalausgangsprozessdaten
1.3.6.1.4.1.13576.10.1.50.5	wioDigitalInLength	R	Länge der Digitaleingangsprozessdaten
1.3.6.1.4.1.13576.10.1.50.6	wioDigitalOutOffset	R	Offset der Digitalausgangsprozessdaten
1.3.6.1.4.1.13576.10.1.50.7	wioDigitalInOffset	R	Offset der Digitaleingangsprozessdaten
1.3.6.1.4.1.13576.10.1.50.8	wioModuleTable	-	
1.3.6.1.4.1.13576.10.1.50.8.1	wioModuleEntry	-	
1.3.6.1.4.1.13576.10.1.50.8.1.1	wioModuleNumber	R	Modulnummer (Slot)
1.3.6.1.4.1.13576.10.1.50.8.1.2	wioModuleName	R	Modulname

Tabelle 443: WAGO-MIB – Process Image Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.50.8.1.3	wioModuleType	R	Modultyp
1.3.6.1.4.1.13576.10.1.50.8.1.4	wioModuleCount	R	Anzahl der Module
1.3.6.1.4.1.13576.10.1.50.8.1.5	wioModule-AlternativeFormat	R	Module in alternativem Format
1.3.6.1.4.1.13576.10.1.50.8.1.6	wioModuleAnalog-OutLength	R	Länge der Analogausgangsdaten des Moduls (Bit)
1.3.6.1.4.1.13576.10.1.50.8.1.7	wioModuleAnalog-InLength	R	Länge der Analogeingangsdaten des Moduls (Bit)
1.3.6.1.4.1.13576.10.1.50.8.1.8	wioModuleDigital-OutLength	R	Länge der Digitalausgangsdaten des Moduls (Bit)
1.3.6.1.4.1.13576.10.1.50.8.1.9	wioModuleDigital-InLength	R	Länge der Digitaleingangsdaten des Moduls (Bit)

16.2.19 Plc Data Group

Die „Plc Data Group“ enthält Werte, die zum Datenaustausch mit dem CoDeSys-Programm verwendet werden.

Tabelle 444: WAGO-MIB – Plc Data Group

Identifier	Eintrag	Zugriff	Beschreibung
1.3.6.1.4.1.13576.10.1.100.1	wioPlcDataTable	-	
1.3.6.1.4.1.13576.10.1.100.1.1	wioPlcDataEntry	-	
1.3.6.1.4.1.13576.10.1.100.1.1.1	wioPlcDataIndex	R/W	Nummer der PLC-Daten (DWORD)
1.3.6.1.4.1.13576.10.1.100.1.1.2	wioPlcDataReadArea	R/W	Lesbare PLC-Daten (DWORD)
1.3.6.1.4.1.13576.10.1.100.1.1.3	wioPlcDataWriteArea	R	Schreib-/Lesbare PLC-Daten (DWORD)



Abbildungsverzeichnis



Tabellenverzeichnis













WAGO Kontakttechnik GmbH & Co. KG
Postfach 2880 • D-32385 Minden
Hansastraße 27 • D-32423 Minden
Telefon: 05 71/8 87 – 0
Telefax: 05 71/8 87 – 1 69
E-Mail: info@wago.com

Internet: http://www.wago.com

